See 1473

B.S.

COMPUTER SCIENCE TECHNICAL REPORT SERIES

UNIVERSITY OF MARYLAND
COLLEGE PARK, MARYLAND
20742

1

ALGORITHMS AND HARDWARE TECHNOLOGY FOR IMAGE RECOGNITION

Semi-Annual Report
1 May - 31 October 1976

Contract DAAG53-76C-0138
(DARPA Order 3206)

Computer Science Center
University of Maryland
College Park, MD 20742

ABSTRACT

Techniques have been developed for detecting and extracting object-like regions from FLIR images. These regions are classified into several target and non-target categories based on their statistical and structural properties. The intent is to develop a methodology and a hardware technology to cue proposed target regions in real time.

TABLE OF CONTENTS

Preface

1. Overview

2. Modelling of Target Scenes

   2.1 Introduction
   2.2 The noise-free case in one dimension
   2.3 Extension to two dimensions
   2.4 Extension to noisy scenes
   2.5 Extension to Markov fields
   2.6 Segmentation procedures based on the model

3. Noise Reduction by Median Filtering

   3.1 Mean filtering
   3.2 Median filtering
   3.3 Algorithms
   3.4 Iterated filtering

4. Object Detection

   4.1 Spot detection
   4.2 Boundary point detection
   4.3 Corner detectors

5. Threshold Selection

   5.1 A median edge operator
   5.2 Prediction of gradient cutoff p-tile
   5.3 Non-maximum suppression of edge responses

6. Post-Processing and Connected Component Extraction

   6.1 Further experiments with shrink/expand
   6.2 Noise component rejection based on lack of edge evidence

7. Feature Selection and Target Classification

   7.1 General approach
   7.2 Feature selection
   7.3 Classification scheme
   7.4 Feature reduction
   7.5 Classifier selection, tuning and testing

8. Data Bases

   8.1 The NVL data base
   8.2 The Alabama data base
   8.3 The sequential data base

9. Plans for the Second Semi-Annual Reporting Period

9.1 Image modelling
9.2 Design of a comprehensive system
9.3 Cooperating sources of information
9.4 Object extraction
9.5 Feature evaluation and target classification
9.6 Hardware constraints

10. References

Preface

This report serves two functions. As a semi-annual report, it records our current understanding of the scene analysis task and the hardware implementation required for the development of a smart sensor. Section 1.1 is an overview of the algorithmic approach being pursued. In Section 9, we discuss areas where future efforts are planned. The Westinghouse portion of this report deals with the hardware technology required to implement the candidate algorithms.

The other function of this report is to discuss the accomplishments of the second quarterly report. Sections 2-8 detail the variety of efforts which have been devoted to the smart sensor project, and an evaluation of their success.

1. Overview

Our approach to target cueing involves two tasks -- object extraction and object classification. Object extraction is the aggregation of image points into disjoint sets by virtue of their position, gray level, or relationship to neighboring points. Object classification is the labelling of these point sets with names from a pre-existing collection based on information known in general about the scene and derived from each point set.

Currently, extraction is accomplished by thresholding a smoothed (median-filtered) version of the original image. The purpose of smoothing is to increase the correlation between the gray value and the position of a pixel (in the 2-D projection of the 3-D scene), since noise (salt and pepper or ringing) tends to decorrelate the sensor signal from its position. Thresholding seems appropriate as a segmentation technique since targets tend to be the hottest (or occasionally, coldest) objects in a scene and thus correspond to extremal values of sensor response. Moreover, in our data bases, objects tend to radiate more or less equally over their surfaces. Our approach would fail if some target class did not correspond to a primitive region (i.e., consisted of pieces) or if a target could be simultaneously hotter and colder than ambient. The assumptions which we have made appear to be valid for the given data bases and for any environment which emphasizes small target acquisition.

An automatic threshold is currently computed based on the heuristic that the edges surrounding a target are the strongest (most contrasting) edges in the picture. There have been exceptions to this. However, in those cases the edges corresponded to a cold region. We are currently not considering such regions, though they could easily be handled analogously.

The result of thresholding is a set of image points (positions). Some of these occur in large clusters while others tend to be small and isolated. A post-processing step eliminates small or thin isolated regions by deleting above-threshold points which have too many below-threshold points as neighbors. Since this deletion is performed simultaneously at all positions, small regions may disappear. Of course, borders of large regions will also be deleted and the regions will shrink. However, these borders can be recovered to some extent by re-expanding (adding those points adjacent to a sufficient number of undeleted points).

After post-processing, the remaining points are aggregated into connected components based on adjacency. This ends the extraction phase. In the classification phase, features describing the component and its relationship to the background are evaluated. Generally, a feature describes a geometric (size, shape) or grayscale (contrast, smoothness) property of the component. By testing the feature values in some order, a decision can be made assigning the component to some labelled class. The construction of a decision algorithm is often a train and test process in which certain features are judged according to their relevance to class identifications. Statistical pattern recognition is a discipline which provides a variety of approaches to this problem.

This overview is meant to provide a framework for understanding the current work. Future work will investigate the relevance of space domain context (a priori knowledge of the scene, scene description) and time domain context (tracking). Progress in these areas will contribute to a deeper understanding of the target cueing paradigm.

In the First Quarterly Report, methods of screening windows to determine whether or not they contain objects were discussed. This approach was not pursued during the second quarter, largely because it was found difficult to model the problem. In any event, a new approach currently under study, based on the coincidence of edges and connected component borders (see Section 9.4), is expected to make a screening step unnecessary.

An overall system block diagram based on the steps outlined above is shown in Figure 1.1. This diagram represents the approaches studied during the past reporting period. Other approaches are also under investigation and are expected to lead to modifications in the system design during the coming period.

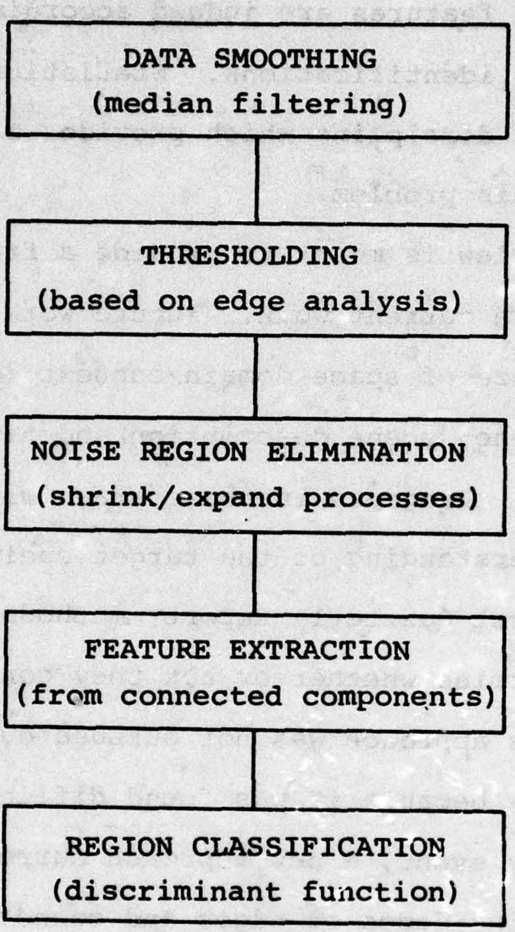


Figure 1.1. Overall system diagram.

2. Modelling of Target Scenes

2.1 Introduction

The statistical modelling and analysis of FLIR imagery presented in this section is an extension of the analysis contained in our first quarterly progress report on this project [1]. The main goal here is to attempt to analyze FLIR imagery on the basis of more realistic assumptions than those in the earlier report. In that report the image modelling was based on first order approximations and assumptions that made the analysis tractable. In particular the "gradient" or "edge" operator, e(i,j), was assumed to be an instantaneous operator of the form

$$e(i,j) = \sqrt{[x(p_1)-x(p_2)]^2 + [x(p_3)-x(p_4)]^2} \qquad (1)$$

where p_1, p_2, p_3, and p_4 were four pixels in the neighborhood of (i,j), and x(·) denoted the noisy gray level observed at the corresponding pixel (see Eqn. (14) of [1]). In the absence of observation noise, gray level is denoted by s(·) instead of x(·). An instantaneous operator as above is extremely sensitive to noise and hence is not used very often in real practice.

It is our intention here to study the statistical characteristics (and the relationship with gray levels) of the response of an edge operator that is more practical than the above mentioned one. The class of edge operators to be studied is defined by

$$e(i,j) = \max[|e_h(i,j)|, |e_v(i,j)|] \qquad (2a)$$

where $e_h(\cdot)$ and $e_v(\cdot)$ are two measures of the horizontal and vertical components of the edge value at (i,j), given by

$$e_h(i,j) = L[x(i+k_1,j)] - L[x(i-k_2,j)] \tag{2b}$$

$$e_v(i,j) = L[x(i,j+\ell_1)] - L[x(i,j-\ell_2)] \tag{2c}$$

L being a linear operator in a local neighborhood, where k_1, ℓ_1, k_2, and ℓ_2 depend on the neighborhood size.

One other assumption made in our earlier report was that in the absence of observation noise an image consists of two constant gray levels, one in the background region and one in the object region. In practice, however, gray levels in a smooth (noise-free) image are not constant. It has been established [2, 3] that gray levels in a smooth image $s(i,j)$ constitute a wide-sense Markov field satisfying the two-dimensional recursive equation.

$$s(i,j) = \rho_v s(i-1,j) + \rho_h s(i,j-1) - \rho_v \rho_h s(i-1,j-1) + \omega(i,j) \tag{3}$$

where ρ_h and ρ_v are horizontal and vertical correlation coefficients, and $\omega(i,j)$ is a sequence of white Gaussian random variables (white noise) of mean $(1-\rho_v)(1-\rho_h)\mu$ and variance $(1-\rho_v^2)\cdot(1-\rho_h^2)\sigma^2$, μ and σ^2 being the mean and the variance of the smooth image.

We shall attempt to study the effect of modelling an image by a Markov field rather than by constant gray levels. For simplicity we assume that $\rho_v = \rho_h \triangleq \rho$. The random field is characterized by three local parameters, such as μ, σ^2, and ρ. In the special case where $\sigma^2 = 0$ and $\rho = 1.0$ the smooth image has a constant gray level. An image with slowly changing gray level will have a correlation coefficient $\rho < 1.0$ and non-zero variance. Observation noise in an image can itself be treated as an image with mean $\mu = 0$ and $\rho = 0$. In general, images will have $0 \leq \rho \leq 1.0$.

The background and the object can be treated as two different scenes, each characterized by its own three local parameters, μ, σ^2, and ρ[3,4]. Thus the values of μ, σ^2, and ρ change as we pass from the background to the object or vice versa. Changes in σ^2 and ρ, the two second-order statistical parameters, imply changes in textural properties of the background and the object. However, the objects encountered in FLIR imagery are of such small sizes that texture patterns, if any, in the object region are not visually discernible. Hence, as a starting point, the parameters σ^2 and ρ will be assumed to have the same values in the background and in the object.

2.2 The noise-free case in one dimension

As we observed in the earlier report [1], under the assumption that an image in one dimension consists of background of gray level s_0 and object(s) of gray level s_1, connected by a ramp edge (Fig. 2.1), an instantaneous one-dimensional edge operator such as

$$e(i) = |s(i) - s(i-1)| \tag{4}$$

(see Eqn. (2) of [1]) has an output such as that shown in Fig. 2.2 Consequently, the two-dimensional joint histogram of gray level and edge value is as shown in Fig. 5 of [1] (reproduced as Fig.2.3 here). Now we shall investigate the response of an operator of the form described in Eqn. (2). In particular, our one-dimensional edge operator is

$$e(i) = |e_h(i)| \tag{5a}$$

$$e_h(i) = \frac{1}{m}\sum_{k=0}^{m-1} s(i+k) - \frac{1}{m}\sum_{k=1}^{m} s(i-k) \tag{5b}$$

where the "window size" m is a fixed but as yet undetermined integer. The response $e_h(i)$ is the difference between the spatial gray level average of m points ahead of (i) (including the point (i)) and that of m points behind (i), as indicated in Fig. 2.4.

Let h(i) be a smoothing or averaging filter of response

$$h(i) = \begin{cases} \frac{1}{m}, & -\frac{m}{2} < i \le \frac{m}{2} \\ 0 & \text{elsewhere} \end{cases} \tag{6}$$

This requires that m be an even integer. Now the response in Eqn. (5b) can be expressed as

$$e_h(i) = \sum_{k=1-m/2}^{m/2} (i + \frac{m}{2} - k)h(k) - \sum_{k=1-m/2}^{m/2} s(i - \frac{m}{2} - k)h(k)$$

$$= s(i + \frac{m}{2}) \circledast h(i) - s(i - \frac{m}{2}) \circledast h(i)$$

$$= [s(i + \frac{m}{2}) - s(i - \frac{m}{2})] \circledast h(i) \tag{7}$$

where $\circledast$ denotes discrete convolution. Equation (7) analytically demonstrates the intuitively obvious fact that the edge response can be obtained by first obtaining an image defined by the gray level difference between two pixels m points apart and then smoothing (or spatially averaging) the resulting picture over m points.

Let us define a new sequence of gray levels y(i) as the "m-th nearest neighbor" (mNN) difference of the original one-dimensional image s(i); specifically,

$$y(i) = s(i + \frac{m}{2}) - s(i - \frac{m}{2}). \tag{8}$$

It is interesting to observe how the mNN difference picture, for a given m, behaves when an object boundary of width w is encountered in the original picture. For brevity, let the last pixel in the background and the first pixel in the object be denoted by "b" and "o", respectively (see Fig. 2.1). Thus the boundary width is

$$w = o-b. \tag{9}$$

It can be easily verified that the response of the mNN difference, when $m \geq w$, is given by

1) $y(i) = 0$ for $i = 1,2,\ldots,b - \frac{m}{2}$;

2) $y(i)$ linearly rises from 0 to d $(= s_1 - s_0)$ with slope d/w over the $w+1$ pixels $i = b - \frac{m}{2}, b - \frac{m}{2} + 1, \ldots, b - \frac{m}{2} + w$ $(= o - \frac{m}{2})$; the slope and number of pixels are the same as in the actual object boundary;

3) $y(i) = d$ for $(1+m-w)$ pixels, $i = o - \frac{m}{2}, o - \frac{m}{2} + 1, \ldots, b + \frac{m}{2}$;

4) $y(i)$ linearly drops from d to 0 with slope $-d/w$ over $w+1$ pixels, $i = b + \frac{m}{2}, b + \frac{m}{2} + 1, \ldots, b + \frac{m}{2} + w$ $(= o + \frac{m}{2})$;

5) $y(i) = 0$ for $i = o + \frac{m}{2}, o + \frac{m}{2} + 1, \ldots$.

Hence, across an object boundary of width w the response of the mNN difference picture $y(i)$ rises from 0 to a maximum $(= d)$ and drops down to 0 again over $w+m+1$ pixels (including the two endpoints where the response is zero). Similarly, it can be seen that when $m < w$

1) $y(i) = 0$ for $i = 1,2,\ldots,b - \frac{m}{2}$;

2) $y(i)$ linearly rises from 0 to d' $(= dm/w)$ with slope d/w over the $m+1$ pixels $i = b - \frac{m}{2}, b - \frac{m}{2} + 1, \ldots, b + \frac{m}{2}$;

3) $y(i) = d'$ for the $1+w-m$ pixels $i = b + \frac{m}{2}$, $b + \frac{m}{2} + 1$, $\ldots, b + w - \frac{m}{2}$ $(= o - \frac{m}{2})$;

4) $y(i)$ drops linearly from d' to 0 with slope $-d/w$

over the m + 1 pixels $i = o - \frac{m}{2}, o - \frac{m}{2} + 1,$

$\ldots, o + \frac{m}{2}$;

5) $y(i) = 0$ for $i = o + \frac{m}{2}, o + \frac{m}{2} - 1, \ldots$.

Some examples of the response y(i) for various combinations of m and w are shown in Fig. 2.5. A few observations are in order here. Firstly, y(i) is symmetric about the center of the object boundary. Secondly, if $|w-m| >> 1$ then the peak response of y(i) covers many pixels. Thirdly, for very large window sizes the peak response (= d) is constant, but for smaller window sizes the peak response (= d') is a function of the boundary width w. Lastly, the number of pixels over which the response rises from 0 to peak is always the smaller of the two quantitites m and w.

The response of the edge operator $e_h(i)$ of Eqn. (7) can now be obtained by convolving y(i) with h(i). This convolution amounts to averaging y(i) over the past $\frac{m}{2}$ points, the present point (i), and the future $\frac{m}{2} - 1$ points -- m points altogether. It may be pointed out here that instead of selecting $e_h(i)$ as we did in Eqn. (5b), if we choose

$$e_h(i) = \text{Median}[s(i+k), k=0,\ldots,m-1] - \text{Median}[s(i-k), k=1,\ldots,m]$$

then in the object boundary region this corresponds to choosing a Kronecker delta as h(i),

$$h(i) = \begin{cases} 1, & i = 0 \\ 0 & \text{elsewhere.} \end{cases}$$

In this case

$$\begin{aligned} e_h(i) &= y(i) \bullet h(i) \\ &= y(i) \bullet \Delta(i) \\ &= y(i) \end{aligned}$$

Thus the edge response for a "median edge detector" is given by $y(i)$ itself.

In the case of averaging $y(i)$ over m pixels there are three types of results of the convolution, corresponding to the three cases $m \geq w$, $2m \leq w$, and $w < 2m < 2w$. In the following we assume, without any loss of generality, that $e_h(i)$ is non-negative, i.e., $e(i) = e_h(i)$. Now if $m \geq w$

1) $e(i) = 0$ for $i = 1,2,\ldots,b-m+1$;

2) $e(i)$ increases quadratically from 0 to $d(w+1)/(2m)$ over the $w+1$ points $i = b-m+1, b-m+2,\ldots,o-m+1$;

3) $e(i)$ increases linearly from $d(w+1)/(2m)$ to $d - d(w-1)/(2m)$ over the $m+1-w$ points (if $m > w$), $i = o-m+1, o-m+2,\ldots,b+1$;

4) $e(i) = e_{max} = d[1 - (w-1)/(2m)]$ for the w points $i = b+1, b+2,\ldots,o$;

5) $e(i)$ drops from e_{max} to 0 over the $m+1$ points $i = o, o+1,\ldots,o+m$, symmetrically to the rising region of $e(i)$; i.e., $e(o+k) = e(b+1-k)$ for $1 \leq k \leq m$;

6) $e(i) = 0$ for $i = o+m, o+m+1,\ldots$.

When $2m < w$

1) $e(i) = 0$ for $i = 1,2,\ldots,1+b-m$

2) $e(i)$ rises quadratically from 0 to $d'(m+1)/(2m)$ over the $m+1$ points $i = b+1-m, b+2-m, \ldots, b+1$;

3) $e(i)$ rises slower than linearly from $d'(m+1)/(2m)$ to $e_{max} = d'$ over the m points $i = b+1, b+2, \ldots, b+m$;

4) $e(i) = e_{max}$ for the $2+(w-2m)$ points $i = b+m$, $b+m+1, \ldots, b+w-m+1$ $(= o+1-m)$;

5) $e(i)$ drops from e_{max} to 0 over the 2m points $i = o+1-m, o+2-m, \ldots, o+m$, symmetrically to the rising portion of $e(i)$;

6) $e(i) = 0$ for $i = o+m, o+m+1, \ldots$.

When $w < 2m \leq 2w$

1) $e(i) = 0$ for $i = 1, 2, \ldots, 1+b-m$;

2) $e(i)$ rises quadratically from 0 to $d'(m+1)/(2m)$ over the $m+1$ points $i = b+1-m, b+2-m, \ldots, b+1$;

3) $e(i)$ rises slower than linearly from $d'(m+1)/(2m)$ to $e_{max} = d'[1-(2m-w)(2m-w-1)/(2m^2)]$ over the $1+w-m$ points $i = b+1, b+2, \ldots, o+1-m$;

4) $e(i) = e_{max}$ for the $2m-w$ points $i = o+1-m$, $o+2-m, \ldots, b+m$;

5) $e(i)$ drops from e_{max} to 0 over the $w+1$ points $i = b+m, b+m+1, \ldots, o+m$, symmetrically to the rising portion of $e(i)$;

6) $e(i) = 0$ for $i = o+m, o+m+1, \ldots$.

Some examples of the response $e(i)$ for various combinations of m and w are shown in Fig. 2.5. It may be observed that the response of this edge operator is quite different from that of the instantaneous operator (see Fig. 2.2). In particular, the region of nonzero response of the edge operator extends to m pixels before and m pixels after the object boundary. Also, only a few pixels in the object boundary achieve maximum edge response, e_{max}; the rest of the points in the object boundary have lower response.

The corresponding two-dimensional joint histograms, $\hat{p}(s,e)$, of gray level and edge values are shown in Fig. 2.6. The general structure of these histograms is evidently different from their counterparts in the case of the instantaneous edge operator. The following is a comparison of the histograms of the new and old edge operators: The peak edge response e_{max} across a particular object boundary is much higher in the new 2-D histograms than was (d/w) in the old histograms. So, effectively, the histogram gets "stretched" in the direction of increasing edge value, e, as a result of using an edge operator of window size m > 1. With this new operator the probability of a nonzero edge value in the background, as well as in the object region, is not zero as it was with the earlier operator. That is,

$$\hat{p}(e>0 \mid \omega_0) \neq 0$$

$$\hat{p}(e>0 \mid \omega_1) \neq 0$$

where ω_0 and ω_1 refer to the background and the object classes, respectively, as in [1]. Also, with the earlier operator, the probability of getting an edge value less than e_{max} in the object boundary was zero, but this is not the case now, i.e.,

$$\hat{p}(e<e_{max} \mid \omega_2) \neq 0$$

where the class ω_2 corresponds to object boundary. In general, the approximate contour of nonzero density may be as shown in Fig. 2.7. Here consideration is given to the fact that an image may contain several object boundaries with various boundary widths, w, while m is fixed over the entire image.

2.3 Extension to two dimensions

As was mentioned in Section 2.1, a two-dimensional scene containing an object will be treated here as two wide-sense Markov fields (one spatially contained within the other) with identical parameter values except for the means. The approach we take here in analyzing such a scene is to assume first that the correlation coefficient ρ is unity; we relax this constraint later. Under this assumption the background and object Markov fields take on their mean (gray level) values s_0 and s_1 respectively. It may also be pointed out here that since the image is two-dimensional the edge operator should now be two-dimensional, as in Eqn. (2). Specifically, the effect of the operator L is to smooth the picture over a two-dimensional (mxm) window. Thus now

$$e_h(i,j) = \frac{1}{m^2} \sum_{k=0}^{m-1} \sum_{\ell=-\frac{m}{2}+1}^{m/2} s(i+k,j-\ell) - \frac{1}{m^2} \sum_{k=1}^{m} \sum_{\ell=-\frac{m}{2}+1}^{m/2} s(i-k,j-\ell) \qquad (10a)$$

$$e_v(i,j) = \frac{1}{m^2} \sum_{k=-\frac{m}{2}+1}^{m/2} \sum_{\ell=0}^{m-1} s(i-k,j+\ell) - \frac{1}{m^2} \sum_{k=-\frac{m}{2}+1}^{m/2} \sum_{\ell=1}^{m} s(i-k,j-\ell) \qquad (10b)$$

This is equivalent to having a smoothing filter

$$h(i,j) = \begin{cases} 1/m^2, & -m/2 < i \le m/2 \text{ and } -m/2 < j \le m/2 \\ 0 & \text{elsewhere} \end{cases} \tag{11}$$

and

$$\begin{aligned} e_h(i,j) &= [s(i+\tfrac{m}{2}, j) - s(i-\tfrac{m}{2}, j)] \bullet h(i,j) \\ &= y_h(i,j) \bullet h(i,j) \qquad (12a) \\ e_v(i,j) &= [s(i,j+\tfrac{m}{2}) - s(i,j-\tfrac{m}{2})] \bullet h(i,j) \\ &= y_v(i,j) \bullet h(i,j) \qquad (12b) \end{aligned}$$

$y_h(i,j)$ and $y_v(i,j)$ being the mNN horizontal difference picture and the mNN vertical difference picture, respectively.

We now draw attention to Eqn. (2a) which defines the edge response at a point (i,j) as the maximum of the absolute values of e_h and e_v. The reason for taking the maximum in Eqn. (2a) is so that at each point (i,j) the output of the edge detector is the more prominent of the horizontal edge e_h and the vertical edge e_v. Suppose there exists a horizontal object boundary (change of gray level is in the horizontal direction only) in a certain region in the scene. The process of computing the mNN horizontal difference $y_h(i,j)$ in this region will yield a response similar to the y(i,j) obtained in the object boundary region in the noisefree one-dimensional case treated in Section 2.2.

Now we can obtain $e_h(i,j)$ by spatially averaging

$y_h(i,j)$ over the appropriate (mxm) window, and this will be the response of e(i,j) in this region. Here we implicitly assume without loss of generality that $e_h(i,j)$ is non-negative, that is, the gray level increases from a low value to a high value across the object boundary. Similarly, in a region containing a vertical object boundary, e(i,j) can be obtained via $y_v(i,j)$. The point being made here is that the response of the two-dimensional edge operator across an object boundary can be approximated by the corresponding one-dimensional edge operator operating in the appropriate direction. Thus, just as the response of the one-dimensional edge operator spreads over m pixels before and m pixels after the actual object boundary in a one-dimensional scene, the response of the two-dimensional edge operator will spread in the horizontal direction at horizontal object boundaries and in the vertical direction at vertical object boundaries. Moreover, just as in the one-dimensional case, the edge response will achieve some maximum value near the center of the object boundary and gradually taper off toward the background and the object. Hence, the joint histogram should have approximately the same structure as in the one-dimensional case.

Because of the spreading of the edge response there is now a set of background and object points near each object boundary that will have nonzero edge value. Let this set of points in the background be denoted by ω_0', and let the similar points in the object be denoted by ω_1'. For an object of convex shape the probability of finding such regions in the background is higher than that in the object, because the object boundary has more neighboring points in the background than in the object. This will bias the histogram (in the nonzero edge portion) toward the background. Fig. 2.8 shows a sketch of a plausible joint histogram.

2.4 Extension to noisy scenes

The presence of observation noise (assumed to be additive white Gaussian with variance σ_n^2) creates a probabilistic distribution of gray level in the background as well as in the object. That is, the background and the object gray levels, instead of being constant, are normally distributed with common variance σ_n^2 and means s_0 and s_1, respectively. In addition, assuming that gray level and edge values are independent, the conditional distribution of edge values given a certain gray level in the background (or in the object) is a mixture of the edge value distribution due to region ω_0' (or ω_1') and that due to noise.

Let the noisy image be

$$x(i,j) = s(i,j) + n(i,j) \tag{13}$$

where $s(\cdot)$ and $n(\cdot)$ are the smooth image and noise respectively. Thus the conditional pdf

$$p(x|\omega_0) \sim N(s_0,\sigma_n^2) \tag{14a}$$

and the pdf of mNN difference y_h or y_v is

$$p(y|\omega_0) \sim N(0,2\sigma_n^2) \tag{14b}$$

since y is the difference of two normal random variables. The effect of averaging over m^2 sample points (convolving with $h(\cdot)$) is to reduce the variance by m^2. Therefore,

$$p(e_h|\omega_0) = p(e_v|\omega_0) \sim N(0,2\sigma_n^2/m^2) \tag{14c}$$

If

$$e(i,j) = |e_h(i,j)|$$

then

$$p(e|\omega_0) \sim N(0,2\sigma_n^2/m^2)x2, \; e > 0 \qquad (14d)$$

and if

$$e(i,j) = |e_v(i,j)|$$

then also

$$p(e|\omega_0) \sim N(0,2\sigma_n^2/m^2)x2, \; e > 0. \qquad (14e)$$

Assuming equal probability of a horizontal or vertical edge in the background, we have

$$p(e|\omega_0) \sim N(0,2\sigma_n^2/m^2)x2, \; e > 0. \qquad (14f)$$

By a similar argument

$$p(x|\omega_1) \sim N(s_1,\sigma_n^2) \qquad (15a)$$

$$p(e|\omega_1) \sim N(0,2\sigma_n^2/m^2)x2, \; e > 0 \qquad (15b)$$

These class conditional edge value pdf's are due to the contribution of noise only. Similar conditonal pdf's due to the regions ω_0' and ω_1' were shown in Section 2.3. The mixture of the two, which is the actual class conditional edge pdf, is shown in Fig. 2.9. In the object boundary region the pdf of gray level in the presence of noise is that of the smooth image convolved with the noise pdf, and is shown in Fig. 2.10.

Now, in the smooth picture, let μ_z be the edge value (in the object boundary) given a certain gray level z. This value is easily obtained from the joint histogram, Fig. 2.8. By an argument similar to that in Eqn. (14) we have

$$p(e|\omega_2,z) \sim [N(\mu_z,2\sigma^2/m^2) + N(-\mu_z,2\sigma^2/m^2)], \; e > 0$$

$$= \frac{m}{\sqrt{\pi}\, 2\sigma} [\exp\{-\frac{m^2}{4\sigma^2}(e-\mu_z)^2\} + \exp\{-\frac{m^2}{4\sigma^2}(e+\mu_z)^2\}], \quad e > 0. \qquad (16)$$

This is shown in Fig. 2.10.

2.5 Extension to Markov fields

We now relax the constraint imposed on the Markov fields in Section 2.3. We let

$$\sigma^2 \neq 0, \text{ and}$$
$$\rho < 1$$

where σ^2 is the variance of the wide-sense Markov fields $s(i,j)$ corresponding to the background and the object. The class conditional pdf of $x(i,j)$, which is the sum of two independent normal variates $s(i,j)$ and $n(i,j)$, is

$$p(x|\omega_0) \sim N(s_0, \sigma^2+\sigma_n^2) \tag{17a}$$

$$p(x|\omega_1) \sim N(s_1, \sigma^2+\sigma_n^2) \tag{17b}$$

The corresponding pdf of ω_2 (the object boundary region) is assumed to remain unaffected.

The mean of the mNN difference picture remains unchanged, while its variance is

$$\begin{aligned}\mathrm{Var}(e_h|\omega_0) &= E[\{s(i+\tfrac{m}{2},\ j) + n(i+\tfrac{m}{2},\ j) - s_0\} \\ &\quad - \{s(i-\tfrac{m}{2},\ j) + n(i-\tfrac{m}{2},\ j)-s_0\}]^2 \\ &= 2\sigma^2 + 2\sigma_n^2 - 2\sigma^2\rho^m \\ &= 2\sigma_n^2 + 2\sigma^2(1-\rho^m)\end{aligned} \tag{18}$$

which is an increase over the value $2\sigma_n^2$ in Eqn. (14c). Equation (18) also applies to the class ω_1. In the object boundary region ω_2, assuming that the cross-covariance between the object and the background gray level is zero, i.e.,

$$E[\{s(i,j)-s_0\}\{s(k,\ell)-s_1\}\,|\,(i,j) \in \omega_0,\ (k,\ell) \in \omega_1] = 0$$

we obtain as in Eqn. (18)

$$\mathrm{Var}(e_h|\omega_2) = 2\sigma_n^2 + 2\sigma^2.$$

The response of the edge operator is $|e_h(i,j) \bullet h(i,j)|$ or $|e_v(i,j) \bullet h(i,j)|$. Let $e_h(i,j) \bullet h(i,j)$ be denoted by $e_{hm}(i,j)$, m denoting the average over an (mxm) window; we similarly define $e_{vm}(i,j)$ from $e_v(i,j)$. Since e_h and e_v are assumed normal, so are e_{hm} and e_{vm}; but the variance of e_{hm} (or e_m) now depends on the correlation function of e_h (or e_v). If $R_e(i,j)$ is the correlation function of e_h (or e_v) then the variance of e_{hm} or e_{vm}) is the volume under the function obtained by the product of $R_e(i,j)$ with a pyramid of width 2m and height $1/m^2$; the (2mx2m) pyramid function is due to the (mxm) window. If e_h (or e_v) is assumed uncorrelated, i.e., $R_e(i,j) = \delta(i,j)$, then the effect of the pyramid is only to reduce the variance by $1/m^2$. For any other correlation function the variance will be higher. We shall assume here that the variance of e_{hm} (or e_{vm}) is $(1/m^2)$ times the variance of e_h (or e_v). Thus

$$p(e|\omega_0) = p(e|\omega_1) \sim N(0,\ \sigma_0^2/m^2)x2,\ e > 0, \qquad (19a)$$

where $\sigma_0^2 = 2\sigma_n^2 + 2\sigma^2(1-\rho^m)$, and

$$p(e|\omega_2,z) \sim N(\mu_z,\sigma_z^2/m^2) + N(-\mu_z,\sigma_z^2/m^2),\ e > 0 \qquad (19b)$$

$$= \frac{m}{\sqrt{2\pi}\sigma_z}[\exp\{-\frac{m^2}{2\sigma_z^2}(e-\mu_z)^2\} + \exp\{-\frac{m^2}{2\sigma_z^2}(e+\mu_z)^2\}],\ e > 0$$

where $\sigma_z^2 = 2\sigma_n^2 + 2\sigma^2$. Thus the effect of treating the background and the object as two Markov fields rather than two constant gray levels is to increase the variance of the different variates as indicated above. Fig. 2.11 shows the joint histogram of a mixture of two noisy Markov fields.

2.6 Segmentation procedures based on the model

How does the present analysis compare to the previous analysis, reported in [1]? Firstly, we note from the two-dimensional noise-free case that even though the new edge operator is of finite window size, as opposed to the instantaneous operator of the earlier analysis, the joint histogram can still be identified as a mixture of three unimodal densities (see Fig. 2.8). The same can also be said about the Markov model of the image. Thus the segmentation procedures suggested by the earlier analysis, such as thresholding based on the conditional pdf for low edge values, the conditional pdf for high edge values, and valley seeking in the joint pdf, are still valid. As an alternative to valley seeking one may also consider mode seeking or clustering. Besides these, there now exists a new segmentation procedure as described below. As seen in Sections 2.4 and 2.5, the conditional pdf given a gray level attains its highest value at and near the center of the object boundary. Hence the least upper bound (l.u.b.) of the conditional modes should also yield a good gray level threshold for segmentation. This gray level threshold t is such that

$$\mathrm{mode}\{p(e|t)\} = \underset{z}{\mathrm{l.u.b.}}[\mathrm{mode}\{p(e|z)\}] \qquad (20)$$

where z denotes gray level.

The simplest of all the prospective segmentation procedures is thresholding based on the conditional pdf for high edge values. This procedure has been implemented with fairly good results. The details of the procedure are given in Section 5. Some of the other procedures to be tried out are the ones based on the conditional pdf for low edge values and the l.u.b. of the conditional modes. It is noteworthy that there are two other possible procedures, based on the l.u.b. of the conditional means and the l.u.b. of the conditional sums. These are analogous to the procedure based on the l.u.b. of conditional mode suggested above. These procedures have been tested elsewhere [5, 6]. A major disadvantage of those procedures was the unequal prior probabilities of gray level on which the edge pdf's are conditioned. The procedure based on the l.u.b. of conditional modes is expected to be free of this problem.

Captions for Figures 2.1-2.11

Fig.	Title
2.1	An object boundary in one dimension.
2.2	Response of the instantaneous edge operator to the image in Fig. 2.1.
2.3	Two-dimensional joint histogram of gray level and edge value for the scene in Fig. 2.1 (reproduced from Fig. 5 of [1]).
2.4	Location of the response of the edge operator and corresponding window locations for m=4.
2.5	Response of the mNN difference operator and the edge operator to an object boundary of width w.
	5a) $m=8$, $w=2$ $(m > w)$
	5b) $m=2$, $w=6$ $(2m < w)$
	5c) $m=2$, $w=3$ $(w < 2m < 2w)$
	5d) $m=4$, $w=6$ $(w < 2m < 2w)$
	Legend:
	····· Gray level across the object boundary
	- - - mNN difference across the object boundary
	—— edge response across the object boundary
2.6	Two-dimensional joint histogram of gray level and edge response for various values of m and w.
	6a) $m >> w$
	6b) $m << w$
	6c) $w < 2m < 2w$

Figure captions (cont'd)

Fig.	Title
2.7	Approximate contour of non-zero density.
2.8	Joint histogram of a two-dimensional scene containing an object.
	2.8a) m large w.r.t. w
	2.8b) m small w.r.t. w
2.9	Conditional histograms
	2.9a) Edge response in the background region.
	2.9b) Gray level in the background and object regions.
2.10	Histograms in the object-boundary region.
	2.10a) Marginal gray level histogram.
	2.10b) Conditional edge response histogram.
2.11	Joint histogram of a mixture of two noisy Markov fields.

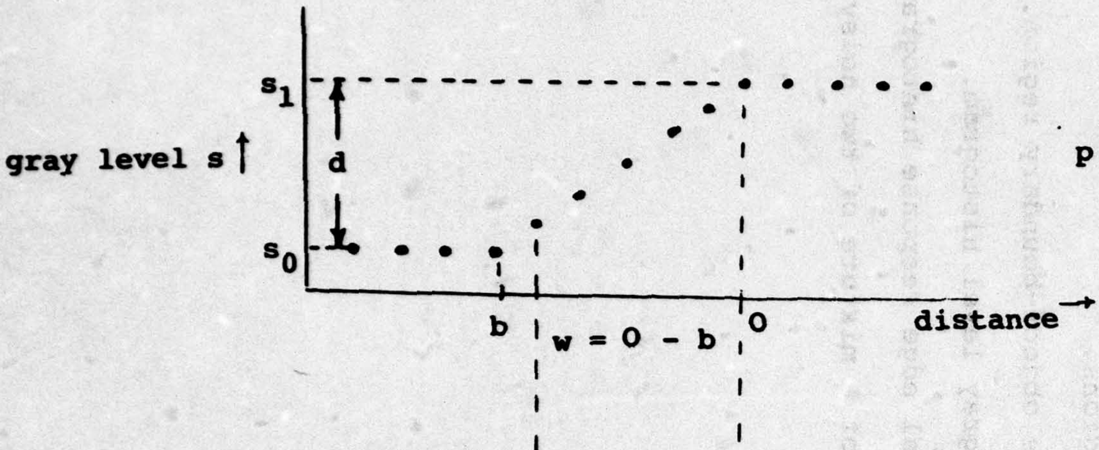


Fig. 2.1

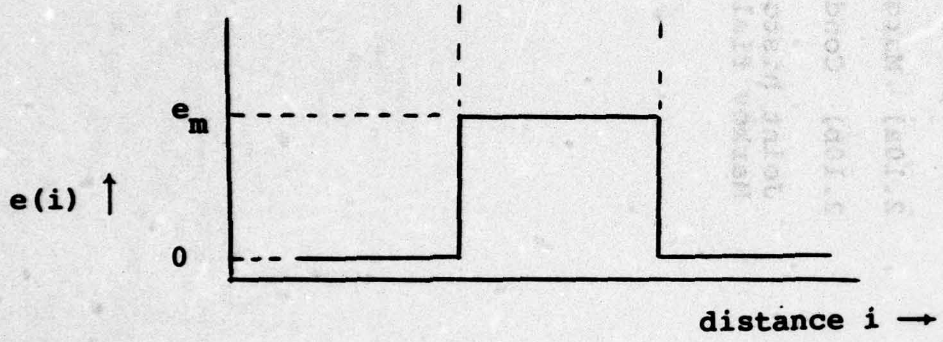


Fig. 2.2

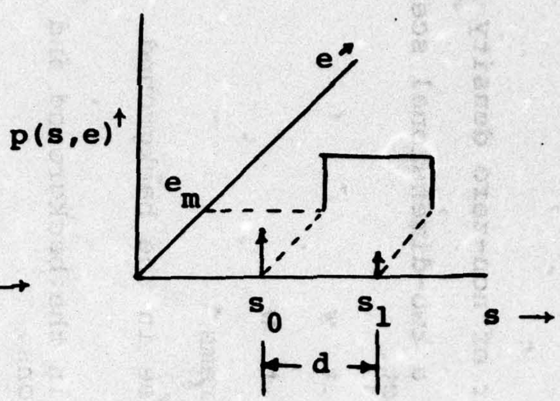


Fig. 2.3

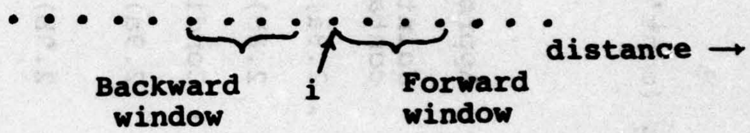


Fig. 2.4

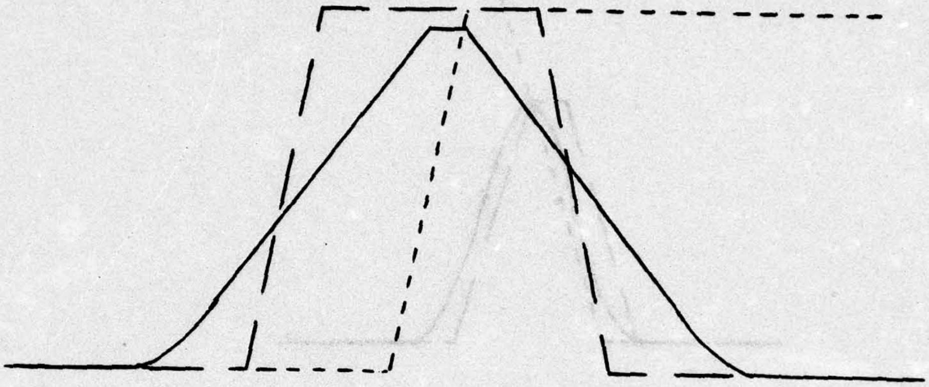


Fig. 2.5a

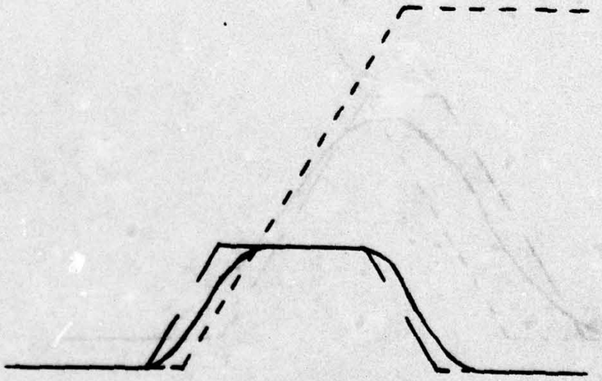


Fig. 2.5b

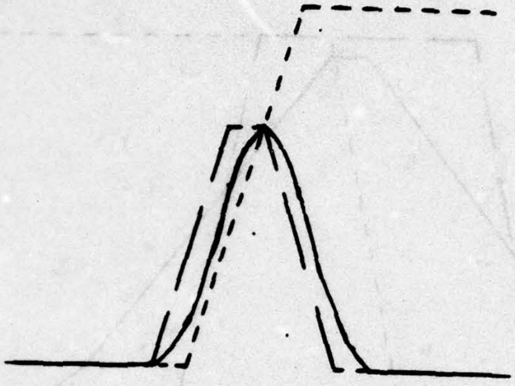


Fig. 2. 5c

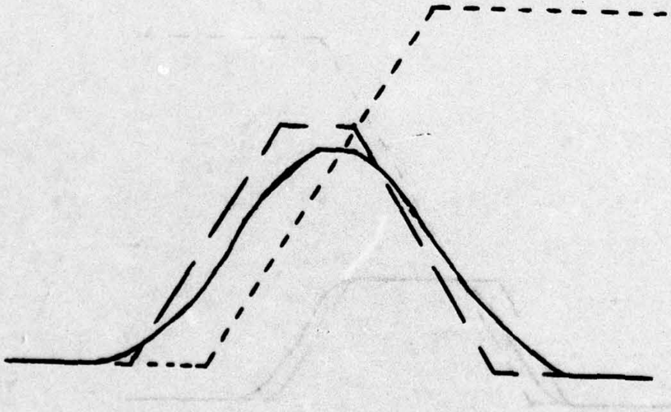


Fig. 2. 5d

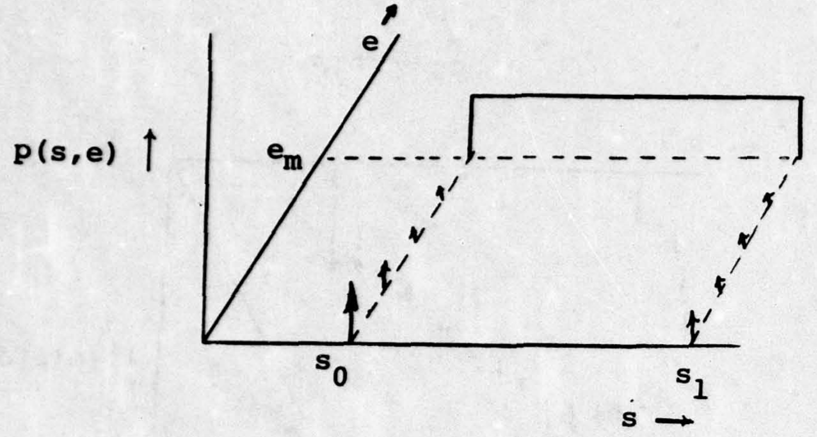


Fig. 2.6a

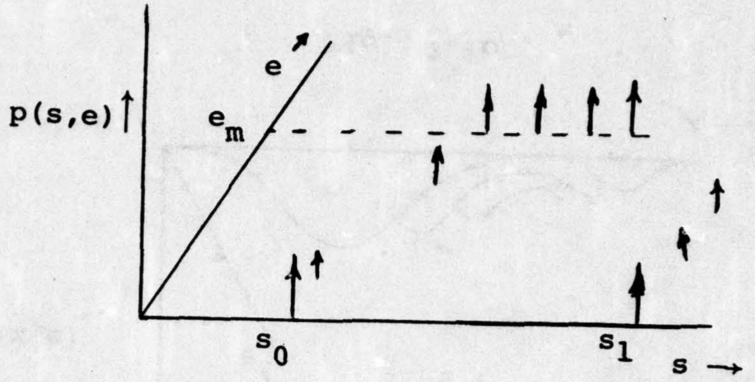


Fig. 2.6b

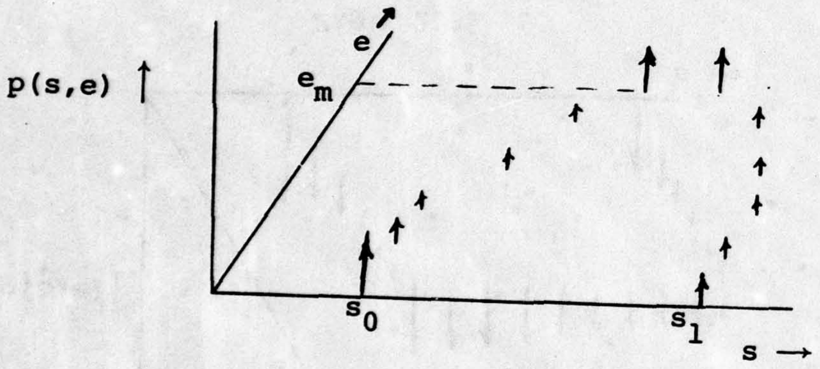


Fig. 2.6c

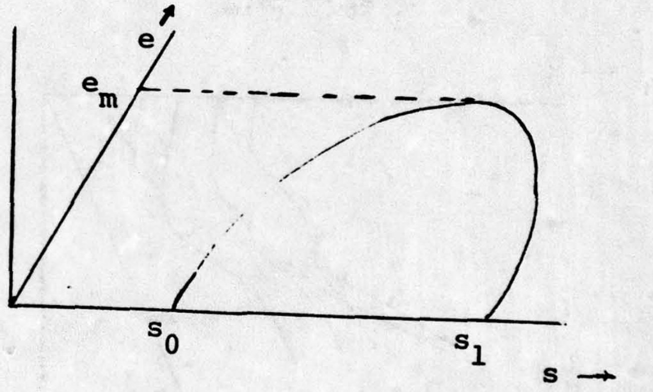


Fig. 2.7

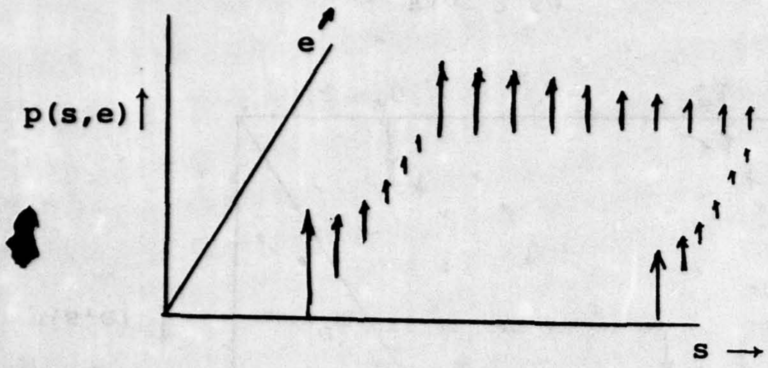


Fig. 2.8a

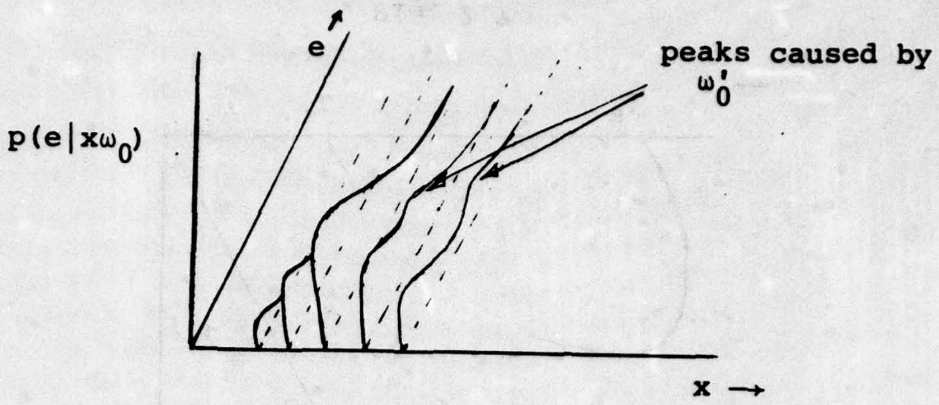


Fig. 2.9a

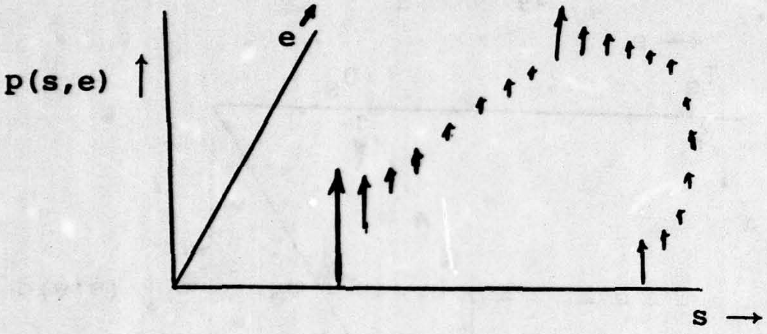


Fig. 2.8b

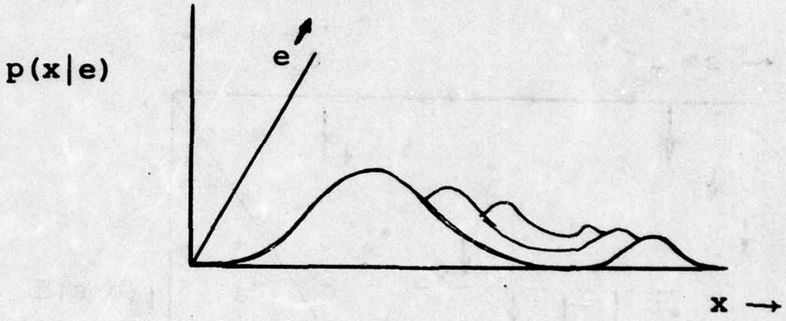


Fig. 2.9b

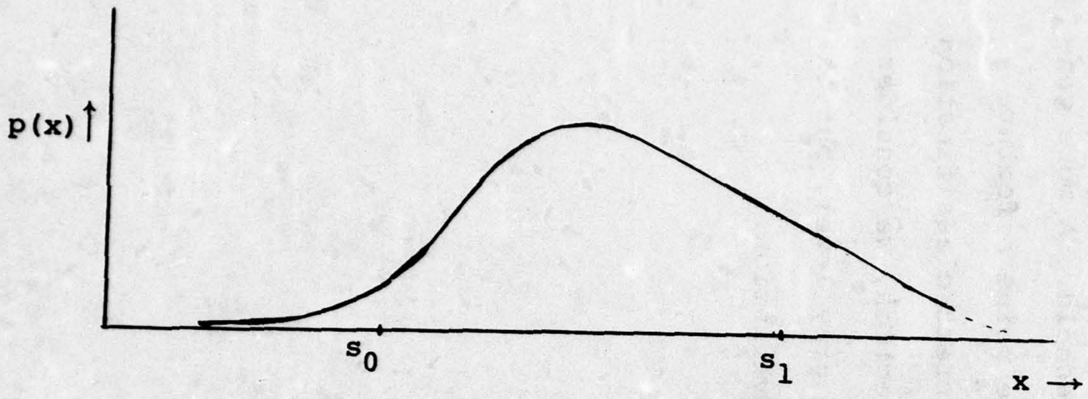


Fig. 2.10a

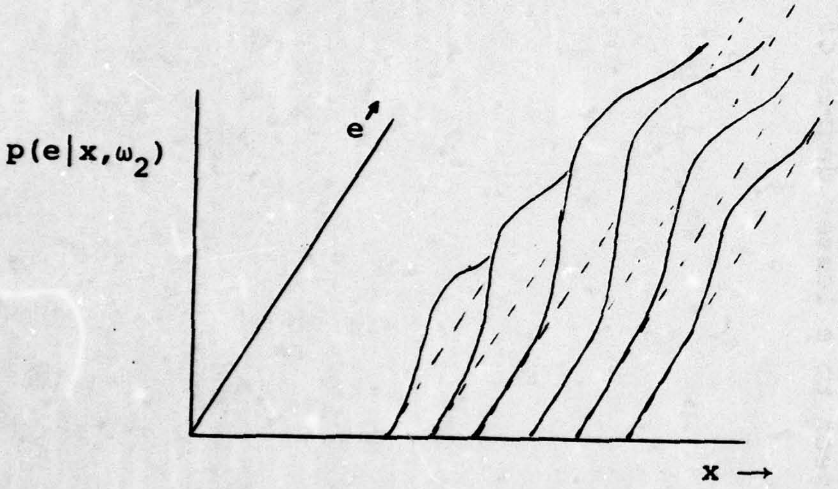


Fig. 2.10b

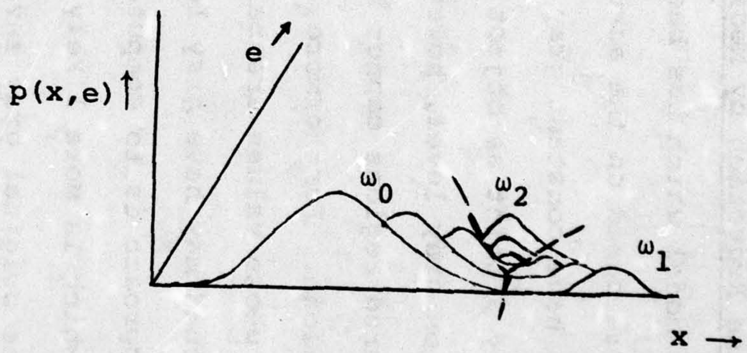


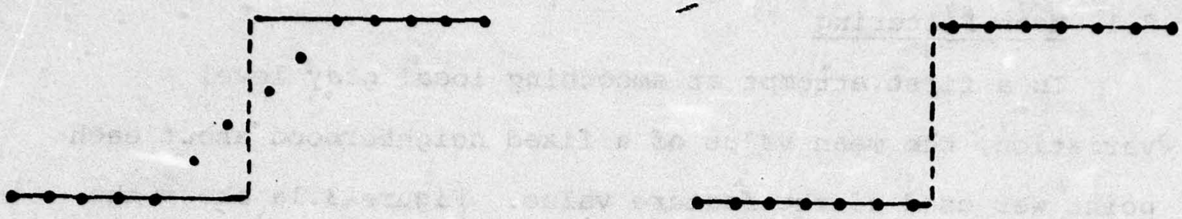
Fig. 2.11

3. Noise Reduction by Median Filtering

The model which has been invoked to account for the FLIR images is based on the assumption that object and background regions have constant gray value plus noise. The decision to classify a point as object or background need not depend solely on gray level, however. In particular, textured or structured regions cannot be segmented on grounds of gray level alone. Furthermore, faint objects are likely to have points whose values are below threshold, while portions of the background may have gray level above threshold. A more general approach is to compute at each image point a feature value which is more likely to yield a correct classification than the original gray level. In this section, we consider two features that are closely related to gray level, but that are subject to a lesser degree of local variation.

3.1 Mean Filtering

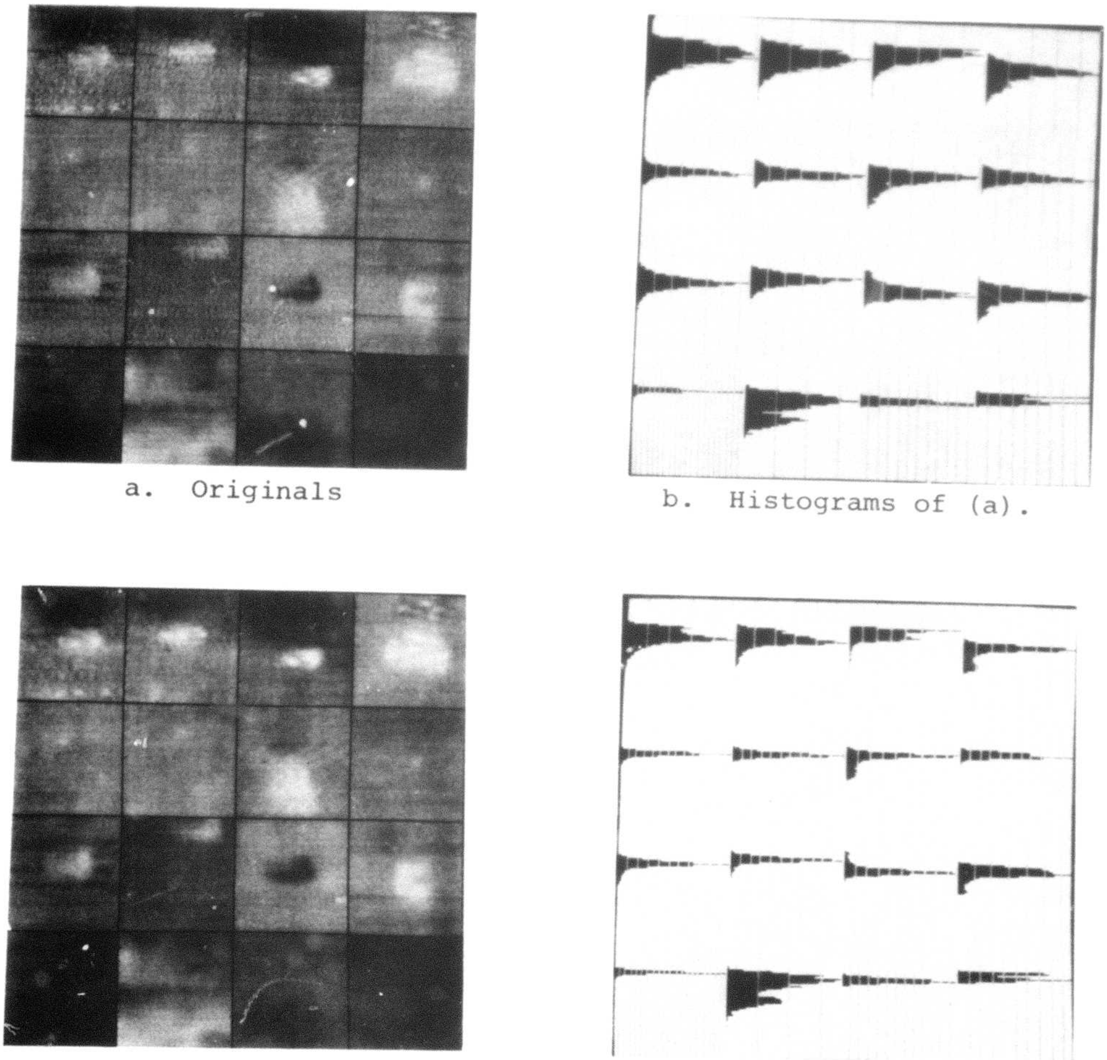
In a first attempt at smoothing local gray level variation, the mean value of a fixed neighborhood about each point was used as the feature value. Figure 3.1a shows the effect of replacing each point of a step edge by its mean value (blurring). Figure 2.5 illustrated blurring for a ramp edge. As is evident, blurring smears edges. Figure 3.2a-d illustrates blurring for several target windows, and also shows the histograms of these windows before and after blurring. Note that blurring tends to blend peaks in histograms, thus making thresholding more difficult. Also, small faint objects tend to become less distinct.



a. Mean filtering

b. Median filtering

Figure 3.1. Effect of filtering on step edges using a five point neighborhood.



a. Originals

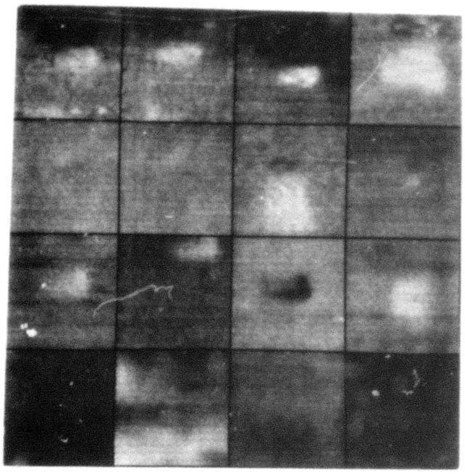
b. Histograms of (a).

c. 3x3 mean filtered windows.

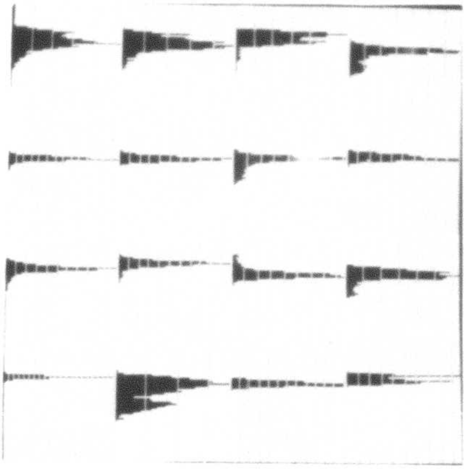
d. Histograms of (c).

Image Reference: 3T 4T 6T 24T
34R 35R 41R 52R
21A 22A 23A 37A
14N 20N 26N 38N

Figure 3 2. Comparison of mean and median filtering.



e. 3x3 median filtered windows.



f. Histograms of (e).

Figure 3.2 (continued)

3.2 Median Filtering

A second approach to image smoothing was investigated, based on median filtering, which has the property of preserving edges. At each point of an image, the median value of the gray levels over a kxk neighborhood is computed. The value of k depends on the amount of local noise variation. For the original images, a 5X5 neighborhood size was chosen. Figures 3.1b and 2.5 illustrate the effects of median filtering on a step and ramp edge, respectively. Note that the median does not increase the ramp width. Thus edges do not smear. In the two-dimensional case, however, corner points are deleted, thereby rounding corners as a side effect. Considering the fuzziness of the boundaries in the data base, this should present no complications. Figure 3.2e-f illustrates a number of median filtered windows and their histograms.

3.3 Algorithms

Efficient algorithms for computing the median of n numbers have been known for some years. The general algorithm for median computation is of order n. However, a better result may be obtained when evaluating a running median. If the data are represented in a balanced binary tree and only k insertions and k deletions are made to the data at each point, then about 2klogn tree operations are required. In the image domain, when taking a running median, we have $n = k^2$. For the 5×5 case, n is 25, k is 5, and at least 50 tree operations are necessary.

If we utilize the fact that the range of data values (gray levels) is fixed, a number of different algorithms are possible. Let d be the dynamic range of the data. Construct a vector of d cells which contains the histogram of the n data points. The median corresponds to the bin containing the n/2th value. Inasmuch as bins can be decremented or incremented in a single step by indexing, the number of operations required on the average (assuming a uniform distribution of gray levels over the range) is 2k for the insertions and deletions and d/2 for the median retrieval search. For the case cited above, the number of operations is 10 + 64/2 = 42. In practice, d may be smaller than 64. Obviously, the quantization range varies for different sensors.

A refinement of this algorithm may give better results for large d (a power of 2), as follows: Construct a binary tree in which the root represents the number of data points with gray levels greater than d/2. Its left son represents the number of data points with gray levels greater than d/4 but not greater than d/2. The right son of the root counts the number of data points greater than 3d/4, etc. The height of this tree is $\log_2 d$. Each insertion or deletion requires the updating of log d entries and the search for the median requires an additional log d steps. The total number of operations required is (2k+1)log d . In the case under consideration, this is 11·6 = 66 operations. Note that for 256 gray levels, the result would be 88 operations.

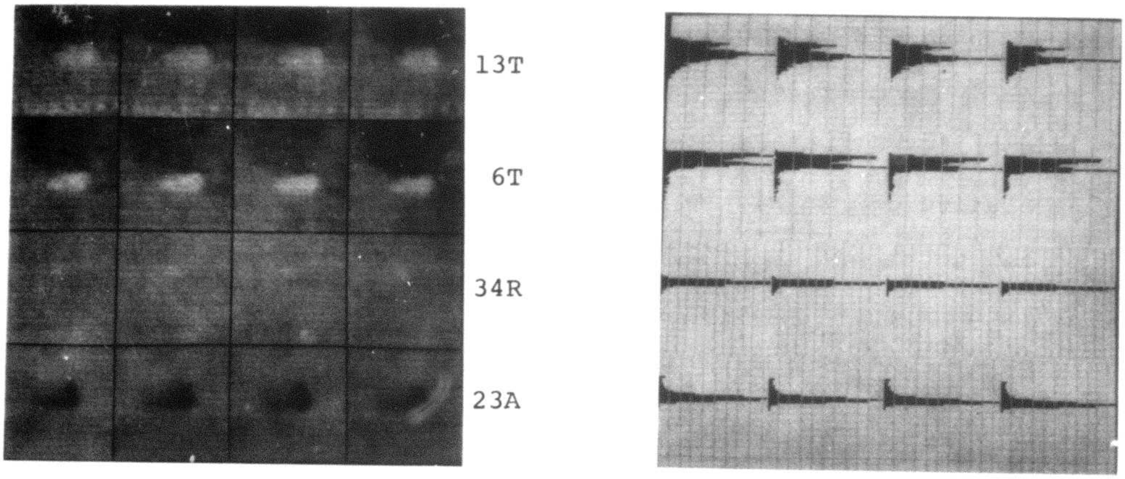
The final method to be described (and the one implemented) makes use of the high autocorrelation of gray level in most images. The cumulative histogram of the n data points is maintained in a vector of length d. The k deletions and k insertions are interleaved in pairs. Each (deletion, insertion) pair isolates a region of the vector which must be modified. The smaller this region on the average, the less work to be done. If the deletion and insertion in a given pair affect the same bin, no change is necessary. The length of the region of change in the cumulative histogram is the expected gray level difference of points at distance k. This corresponds to a variogram value, v(k). After updating, the vector is binary-searched for the median. Thus the number of vector operations is k·v(k), followed by log d operations to binary-search the updated vector. The sum k·v(k) + log d should be quite small for relatively smooth images.

3.4 Iterated Filtering

As mentioned earlier, the effect of median filtering is to reduce local variation. Clearly, any isolated noise spike whose area is less than half the window size will be eliminated. In the one-dimensional case, any waveform which is k-locally monotone (i.e., whose adjacent point value differences do not reverse sign within any region of length k) is left unchanged by median filtering over k sized neighborhoods. Knowledge of the image domain can provide reasonable estimates of appropriate values of k.

Given a waveform, one can converge to a locally monotone smoothing of it by iterating the median filtering process until no further change occurs. There are, however, certain "pathological" waveforms which do not converge. For example, ...1 3 1 3 1 3 1... switches to ...3 1 3 1 3 1 3... and back again when using odd size neighborhoods. One would not expect such configurations in practice.

The notion of local monotonicity does not extend to two dimensions since corner points are subject to deletion by median filtering. The use of cross-shaped rather than square neighborhoods will save corner points but will also enhance thin lines. Iterated median filtering can, however, be used successfully on two-dimensional images. Figure 3.3 shows results of its application to several test windows. Although the images seem to improve somewhat in the process, it was felt that the benefits were marginal.



a. Iterated 3x3 median filtering.

b. Histograms of (a).

Figure 3.3. Effects of iterated median filtering.

Column 1: originals. Columns 2,3,4: iterations 1, 2, 3.

4. Object Detection

In the previous quarterly report, the thresholding study demonstrated that large targets could be isolated automatically. Smaller targets, however, were much more elusive due mainly to the inability of the threshold selector to choose an applicable threshold. Inasmuch as distant targets entering the sensor field of view are likely to be small and faint, and since acquisition of such targets is crucial to the task, a study devoted to detecting such objects was initiated. This section describes several attempts at small object detection.

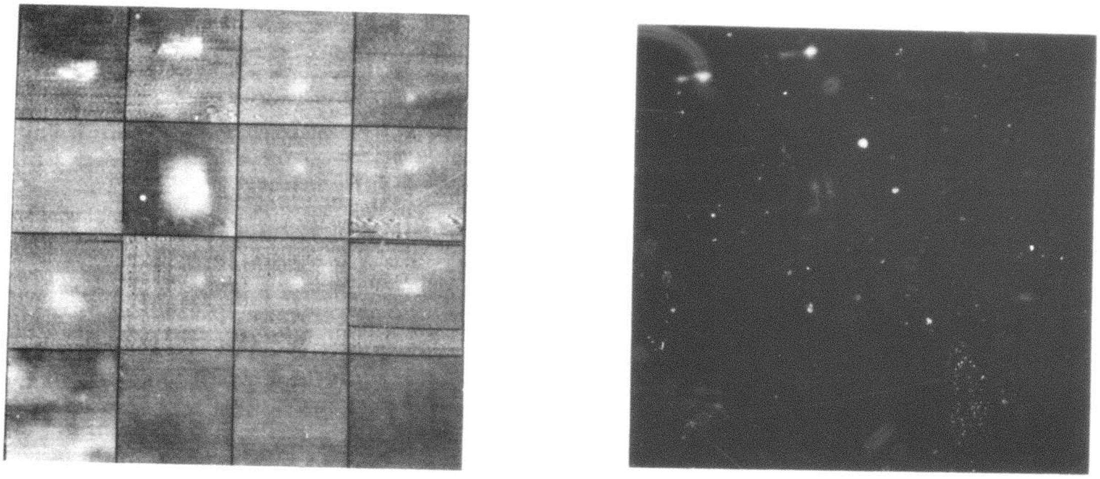
Another aspect of object detection that should be mentioned is the problem of objects that are not entirely contained within a window. For object classification (see Section 7), the features used should be measured over the entire object; but if an object overlaps two windows, the thresholds used to extract it from the windows may differ, and feature measurements made on the pieced-together object may thus be inconsistent. This problem can be avoided by using overlapping windows, and discarding any object that touches a window boundary. If the windows overlap by 50%, and they are at least twice the object size (in each linear dimension), then for any object there will be at least one window that entirely contains it, so that no objects will be lost (except at the edges of the frame) by discarding objects that touch window boundaries.

4.1 Spot Detection

A spot is a localized region of gray level activity which contrasts with its surround. One can construct spot detectors which measure at each point the amount of such activity. One such detector is the generalized Laplacian, defined as the difference in average gray level between an $h \times k$ central rectangle and an $m \times n$ surround. Knowledge of the expected spot geometry can help in choosing the values of h, k, m and n. Figure 4.1 illustrates the results of running generalized Laplacianson test windows. The linearity of the Laplacian operator gives a smeared result which responds to edges as well as to spots. Moreover, a strong edge could provoke more response than a weak spot. Because of this Laplacians are poor spot detectors.

A non-linear operator was devised which attempts to threshold the operator window and compute the percentage of points in the window which have been "correctly" classified, i.e., the percentage of central region points above threshold and of surround points below threshold. If the central region occupies about p% of the window, the threshold can be chosen at the upper p-tile of the gray level histogram. A high percentage of correctly classified points indicates a bright spot; a low percentage, a dark spot; near p%, no spot. Figure 4.2 shows the results of this type of detection procedure. Although non-linear, this detector has an obviously smeared response function.

In summary, both types of spot detector share similar



a. Original windows.

b. Generalized Laplacians
(center = 7x7,
surround = 13x13)

6T	11T	34T	57T
34R	47R	55R	57R
46A	52A	54A	58A
20N	26N	50N	56N

Figure 4.1. Generalized Laplacians

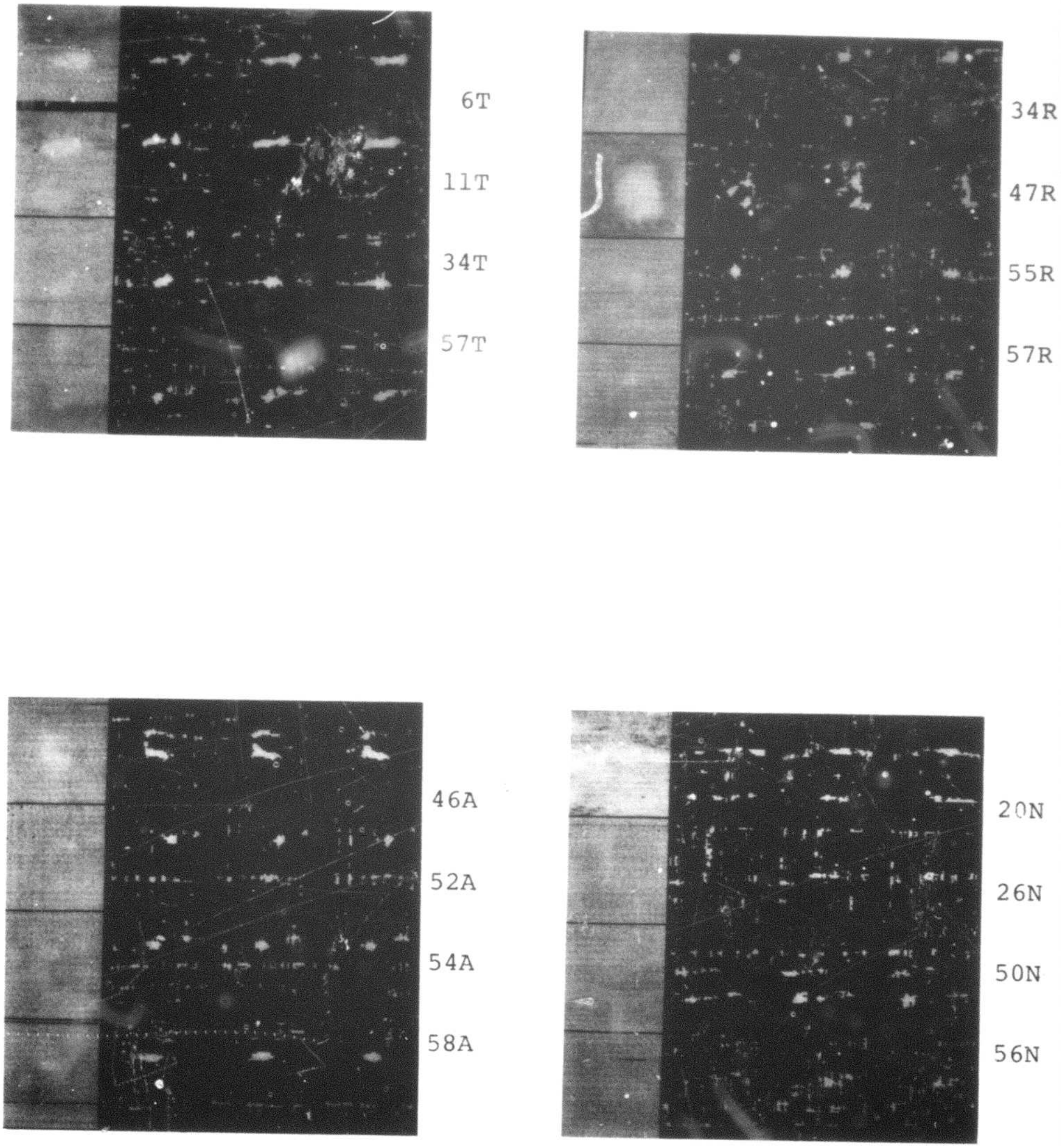
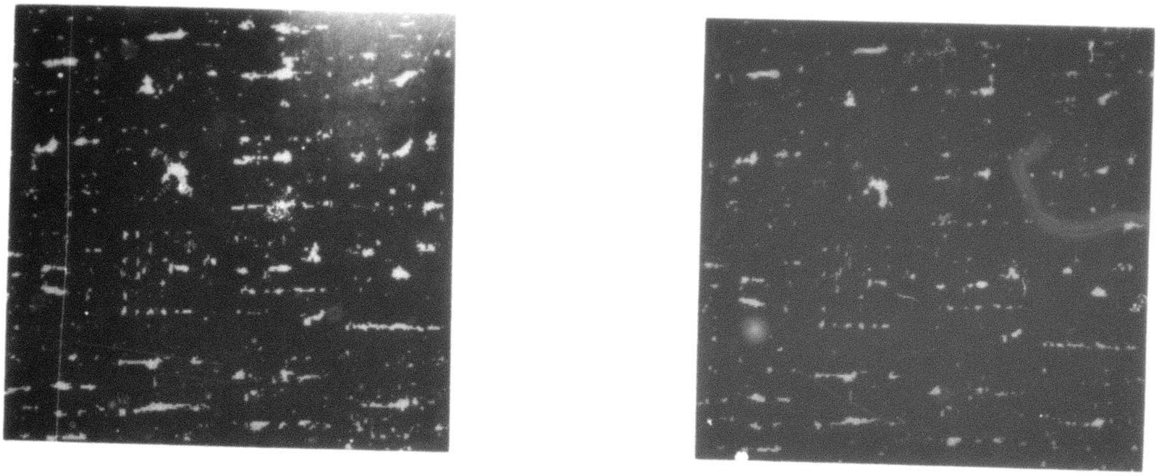


Figure 4.2. Non-linear spot detector. Displayed points are top 5% of responses. Column 1: originals; Columns 2, 3, 4: responses based on square centers of sizes 7, 9 and 11 and square surrounds of sizes 11, 13 and 15, respectively.

problems. Any spot detector which has maximal response for a single "ideal" configuration of points and whose response falls off rapidly for less than ideal configurations is unlikely to ever achieve its maximum response. Moreover, the falloff contributes ambiguity to the decision procedure for near spots. The greatest handicap, though, is the inability to know precisely the shape of the spot to be detected. A square or rectangular central region of fixed size imposes an almost unrealizable constraint. Even if the size constraint is relaxed by choosing the maximum response at a point over a range of sizes (Figure 4.3), the shape constraint limits the effective response. The best alternative is to choose not a template matching approach but a hybrid approach which attempts to assemble a variety of forms of evidence for the presence of a spot.



a. Top 5% of maximum responses.

b. Top 3% of maximum responses.

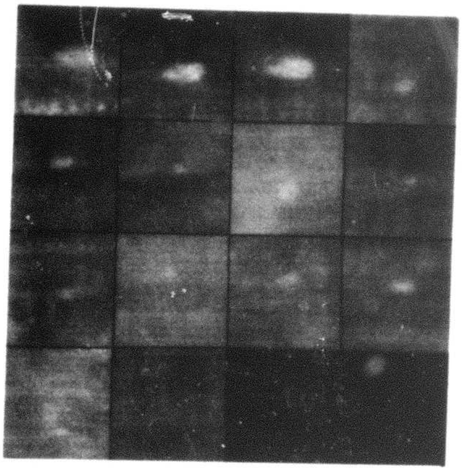
6T	11T	34T	57T
34R	47R	55R	57R
46A	52A	54A	58A
20N	26N	50N	56N

Figure 4.3. A non-linear spot detector which chooses the maximum response at each point over a range of sizes (as in Figure 4.2).

4.2 Boundary Point Detection

As a scan line passes through an object, the leading and trailing edge responses bracket the object. If one could associate the horizontal bracketings from line to line, and similarly for vertical bracketings, the object would be contained within the tracked boundary points. In general, though, it is not known whether an object is contained within a window, and thus whether computed edge responses correspond to object edges; nor is it known whether an object contains spurious or genuine internal edges. Finally, the bracketing assumes that leading and trailing edges both exist on a scan line and have similar properties.

In order to study the seriousness of these problems, a number of images were processed. A 2x4 horizontal difference operator was run on each window and local non-maxima were suppressed. Low response edges (gradient value ≤ 2) were discarded. The resulting edge points are displayed in Figure 4.4 with their orientations and magnitudes. The assumption that the edges separating object from background are the strongest edges seems to be borne out in practice. However, not every left edge has a corresponding right edge, and vice versa. Also, many extra edges appear both within an object and in the background. It appears to be difficult to utilize this information alone in order to extract objects.



3T	6T	8T	34T
54T	58T	31R	52R
58R	34A	54A	58A
2N	26N		

a. Originals.

Figure 4.4. Horizontal boundary points, thresholded at edge value 3. Image values have been multiplied by 4 for visibility.

Figure 4.4 (continued)

b. Horizontal boundary points.

4.3 Corner I

Man-mad

and sharp co

ing them wit

mation to th

was initiate

used is defi

A B
o , wher
C D

ample, a low

A-Max(B,C,D)

are represen

tive 2x2 squ

that type of

similarly fo

illustrates

At each poin

is displayed

was displaye

same orienta

presence of

maximum supp

creased in p

pressed re

thinned res

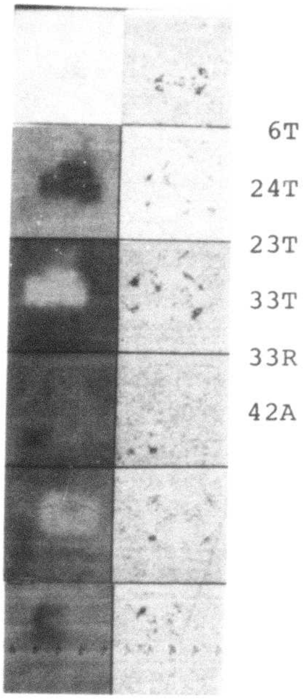


Figure 4.5. Corner detector responses.

Column 1: originals; Column 2: detector responses.

5. Threshold Selection

The method of threshold selection proposed in the first quarterly report has proved successful for almost all of the target windows. A brief description of this method is as follows: Let e(i,j) be the edge value computed at (i,j) and g(i,j), its gray value. Then the chosen threshold is $\overline{g} = \mathrm{AVG}\{g(i,j) \mid e(i,j) > t\}$. This formulation provides several dimensions for variation. This section discusses a number of these.

The choice of operator for computing edge values had been investigated with the conclusion that the difference of 4x4 averages provided superior response to the other tested operators. In Section 5.1, we investigate an edge operator which may provide optimum response for the purposes of threshold selection (although not as a general purpose edge detector). A second parameter in the algorithm is the threshold t. Previously, t was set at the edge value corresponding to the top 20% of edge response. The determination of an appropriate value for t is discussed in Section 5.2. Finally, Section 5.3 investigates the use, in determining g, of only those points whose edge values are local maxima. Such points are likely to lie at the midpoints of edge ramps.

5.1 A Median Edge Operator

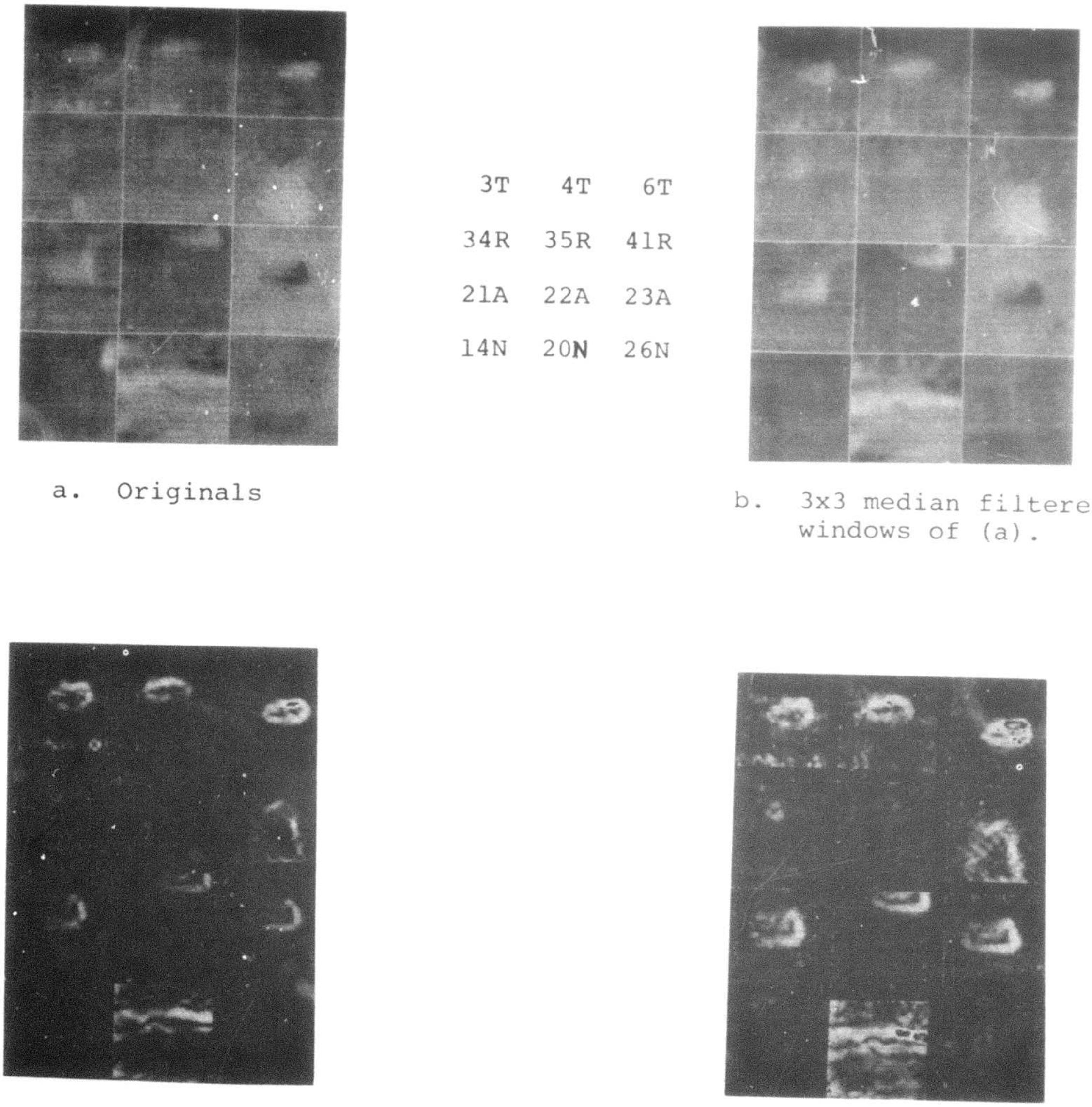
The threshold selection algorithm predicts a threshold based on a sample of image points likely to lie on object/background edges. It has been assumed that object/background edges are the most significant edges in a picture, i.e., that the maximum contrast occurs between objects and background. If it were possible to identify only points lying on these edges, then the threshold selection method would be consistently reliable. Unfortunately, the presence of noise variations and object substructure causes additional points to be added to the sample. Noise variations in the background will cause the threshold to be shifted lower, i.e., toward the mean gray level in the window. Object variations will shift the threshold higher. For the NVL data base, background noise predominates. One would like an edge operator which provides more response at actual edges than at noise edges.

Heretofore, the difference of mean values over adjacent neighborhood has been used to produce the edge responses. As was mentioned in Section 3.1, this operator has maximum response only at the midpoint of an edge ramp. Thus, each edge can provide edge values from zero to the maximum edge response. But as the operator crosses the edge, only a fraction of the edge points provide responses above t. In order to provide a significant sample of edge points, t may have to be fairly low. However, if t is lowered then more and more noise edge points are likely to occupy the sample bins, thus polluting the sample. Thus if we wish to consider p% of the top edge values, we may be forced to choose t to include a bin containing many more points than could possibly arise from object edges. The threshold based on such an overlarge sample is likely to be incorrect (too low). One may improve the sample by increasing the number of above t responses for edge points as the operator crosses an edge.

We have seen in Section 3.1 that the median filter is the identity function on edge ramps. A median difference operator can be shown to have maximum response at all points of the edge ramp. The magnitude of the horizontal difference at (i,j) is $|\hat{g}(i,j-k)-\hat{g}(i,j+k)|$ for k sufficiently large to span the ramp (for our purposes, k = 2 or 3). A vertical difference is defined similarly. The image $\hat{g}$ is obtained by an additional median filtering of the image (see below). Figure 2.5 illustrates the response of this type of operator to a ramp edge, and compares this response to that of the mean difference operator. Note that maximal response is smeared across the edge width. Figure 5.1 illustrates the operator for a selection of test images and compares its response to the mean difference operator.

The effect of the median difference operator should be increase the definition and depth of the U-shaped region of the two-dimensional histogram, thus separating its cusp from the noise values and providing less densely packed bins. Figure 5.2 compares the 2-D histograms for the mean and median difference operators.

Currently, there are no plans to replace the mean



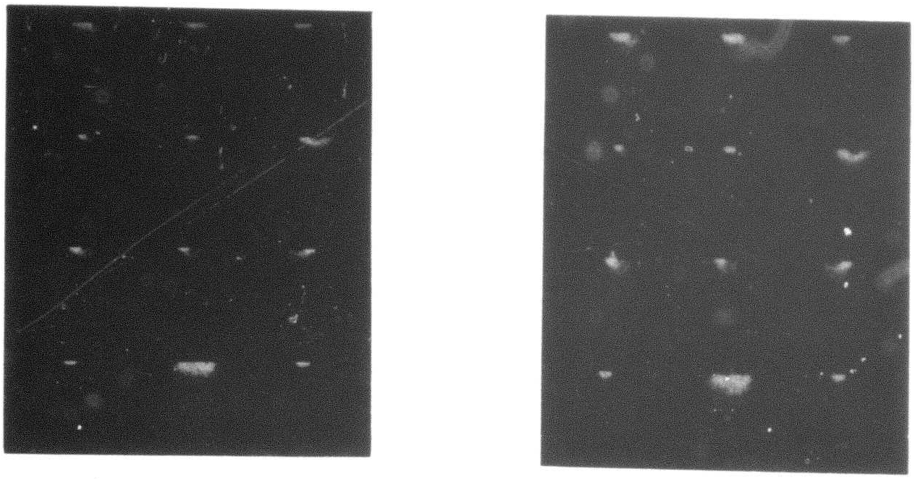
a. Originals

b. 3x3 median filtered windows of (a).

c. Mean difference operator.

d. Median difference operator.

Figure 5.1. Comparison of mean and median difference operators.



a. Using mean difference.

b. Using median difference.

Figure 5.2. 2-D histograms of Figure 5.1.

difference operator by the median, since the median difference fails to localize the edge response. This could be detrimental to later processing which attempts to use edge location as a check on segmentation (Section 5.3). However, from an implementation point of view, hardware already exists for computation of the median at the preprocessing stage. The second median filtering required for differencing would pose no further obstacles.*

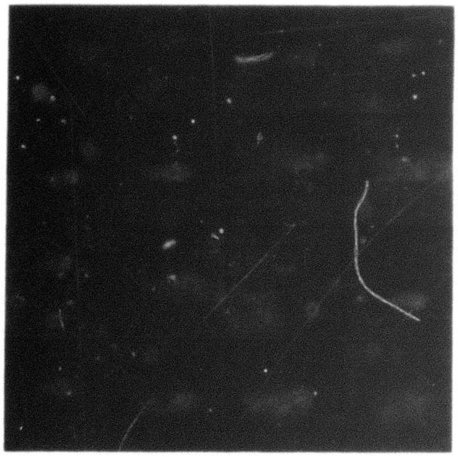
*The first median filtering is performed for purposes of noise smoothing, and the neighborhood size used depends on the noise level. The second median filtering is analogous to the mean computation for edge detection purposes, and uses a neighborhood whose size depends on the widths of the edge ramps. It has been found that poorer edges are obtained if the second filtering is eliminated.

5.2 Prediction of Gradient Cutoff p-tile

Section 5.1 introduced the gradient threshold parameter t. This parameter controls the size and quality of the sample points of high edge value used to compute the gray level threshold. As was mentioned before, setting t too high decreases the statistical reliability of the sample; while a small t may admit too many noise values. Assuming thresholding is to take place (a decision which could be based on the 2-D histogram, for example), the choice of t should depend on the expected amount of object edge. Obviously, many assumptions are built into this notion, e.g., that the window contains only a <u>single</u> object of known <u>size</u>, <u>shape</u>, <u>contrast</u>, <u>resolution</u>, etc. In a tactical situation, one could make estimates of these parameters based on situation data. However, we limit our discussion here to a simple model.

Assume a circular homogeneous target of radius r resolution units on a homogeneous background with edge ramp width w. If a median difference operator is used which computes a difference based on median values k units apart, then the total number of edge response points for the object is about $(2w+k-1)\cdot 2\Pi r$, of which $(w+k-1)\cdot 2\Pi r$ have gradient values above half the contrast. Assuming r ranges from 3-15, $w = 3$, $k = 3$ and the object is within a 64x64 window, the proportion of object edge varies from $50\cdot 3/4096$ to $50\cdot 15/4096$; that is, from 3.6% to 18%. However, if we consider only the values above half the contrast, the proportion ranges from about 2% to 12%. When using the mean difference operator, these values drop somewhat. Clearly, the value of 20% used in the experiments carried out in the first quarter is far too generous. A value of 5% favors smaller objects and has proved adequate in practice (Figure 5.3). A more sensitive model might be able to predict the best percentage value based on the 2-D histogram. For example, if the gray values at gradient value zero separate into two classes of size A_B (background) and A_O (object) then we might guess that r is approximately $\left(\frac{A_O}{A_B} \cdot \frac{4096}{\pi}\right)^{1/2}$ resolution units in a 4096-point window.

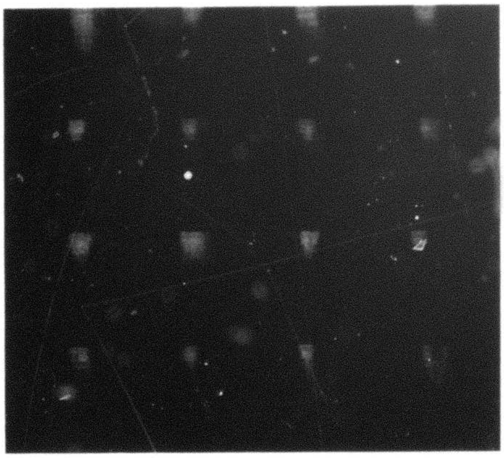
In order to test the sensitivity of the chosen gray level threshold to different percentage values of p, the graph of the threshold was plotted as a function of the gradient cutoff t; see Figure 5.4. There is a tendency for this graph to drift toward the mean gray value as t is decreased (i.e., increasing p). The stability of the chosen threshold for large objects is evident. For small objects, the choice is quite sensitive to the bin size.



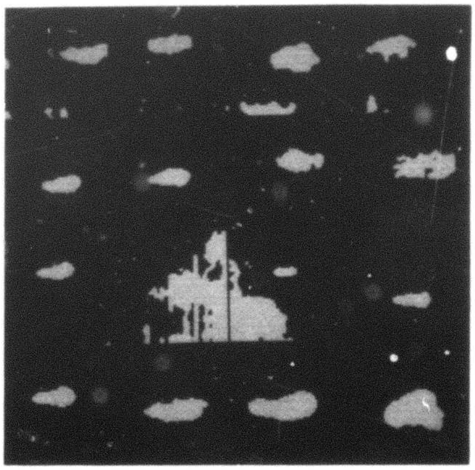
Originals.

1T	2T	3T	4T
6T	8T	9T	10T
11T	12T	13T	14T
15T	16T	17T	21T

Image reference numbers.

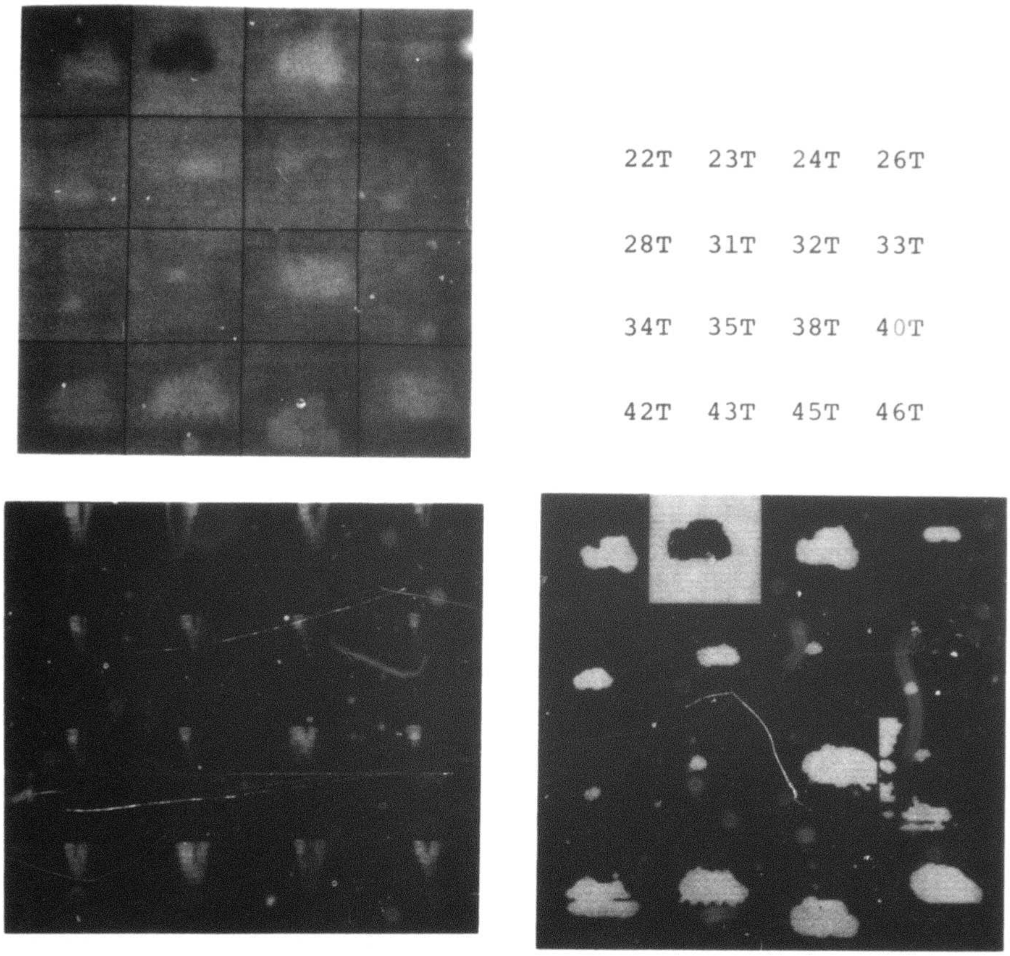


2-D Histograms.



Thresholded windows after shrink/expand.

Figure 5.3. Results of thresholding and post-processing 96 target windows and 10 noise windows.
a. 43 tanks.



22T 23T 24T 26T

28T 31T 32T 33T

34T 35T 38T 40T

42T 43T 45T 46T

Figure 5.3a (continued)

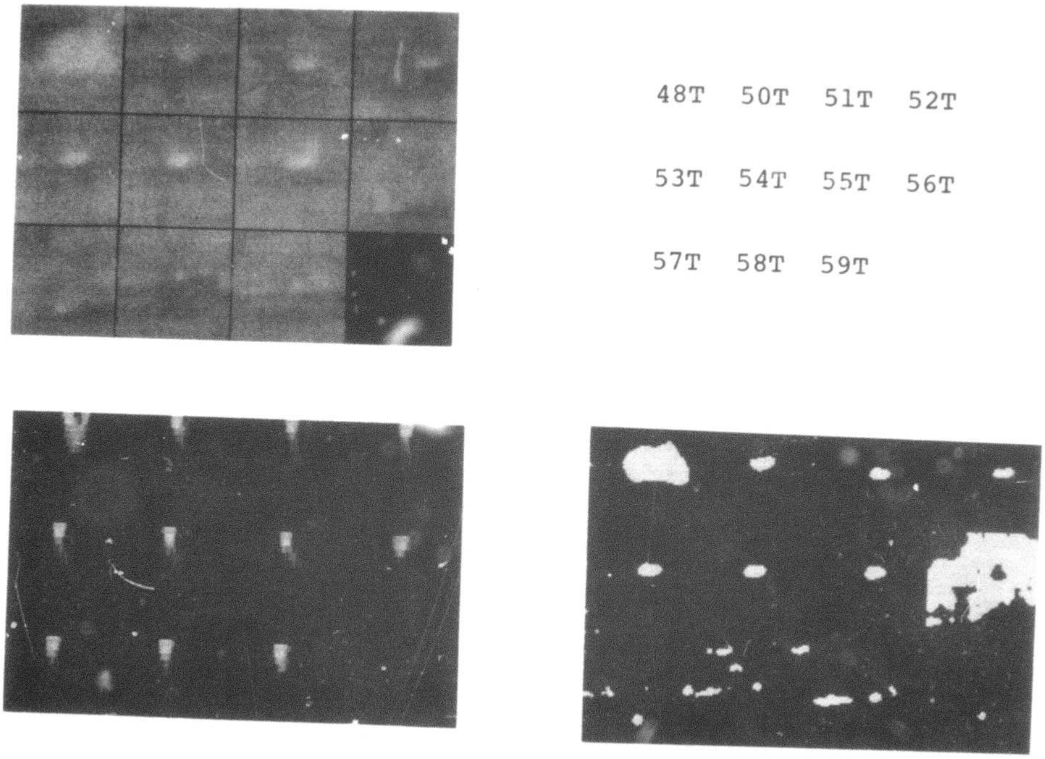


Figure 5.3a (continued)

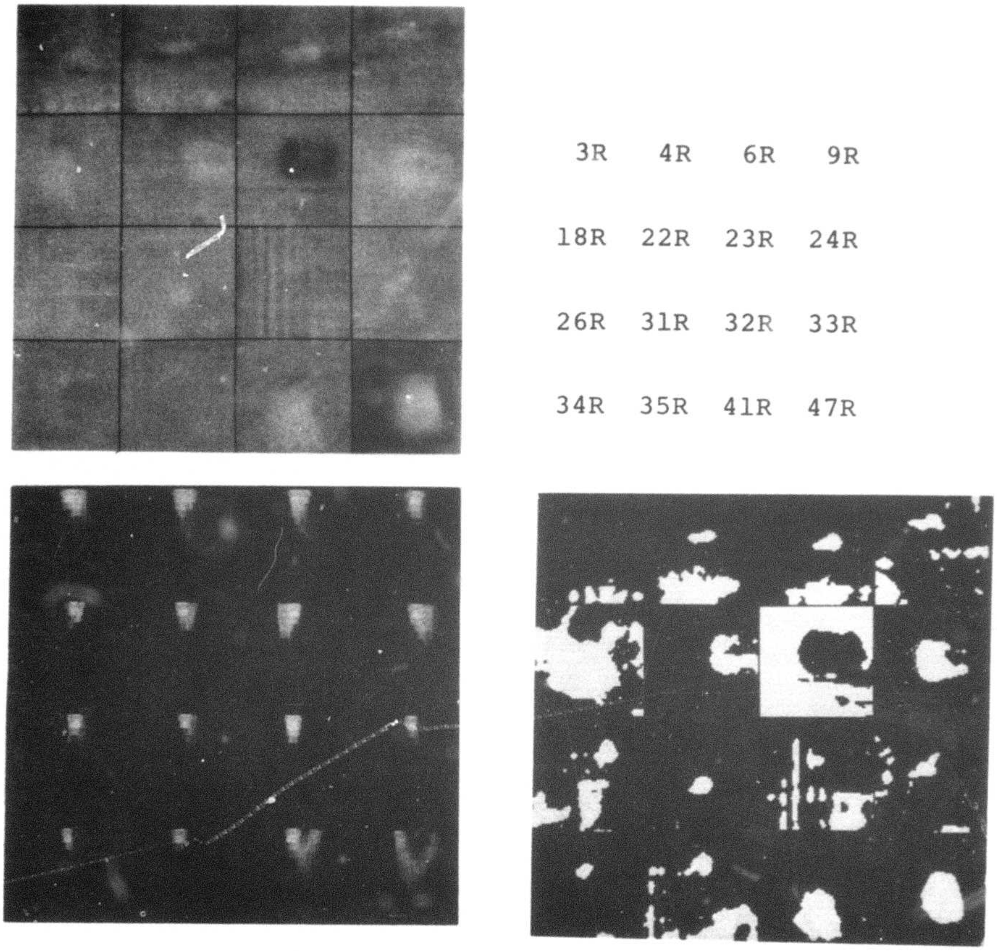


Figure 5.3b. 25 trucks.

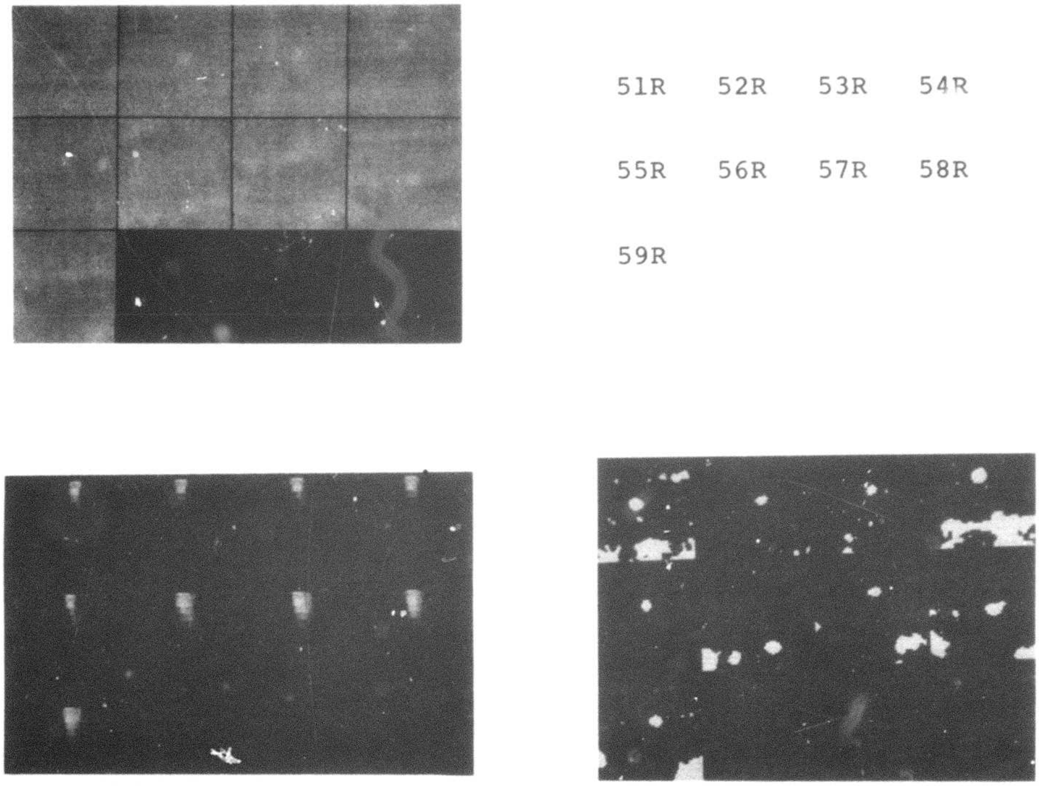
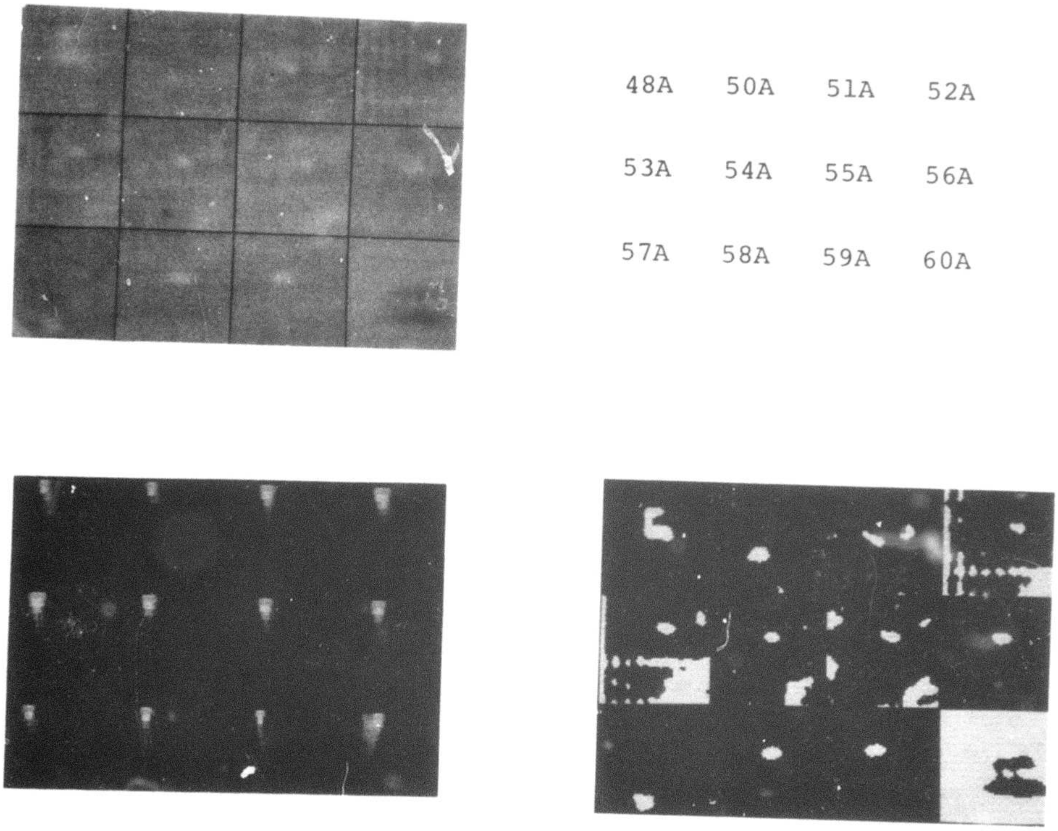


Figure 5.3b (continued)

21A	22A	23A	24A
27A	28A	32A	33A
34A	35A	37A	38A
42A	44A	45A	46A

Figure 5.3c. 28 APCs.



48A 50A 51A 52A

53A 54A 55A 56A

57A 58A 59A 60A

Figure 5.3c (continued)

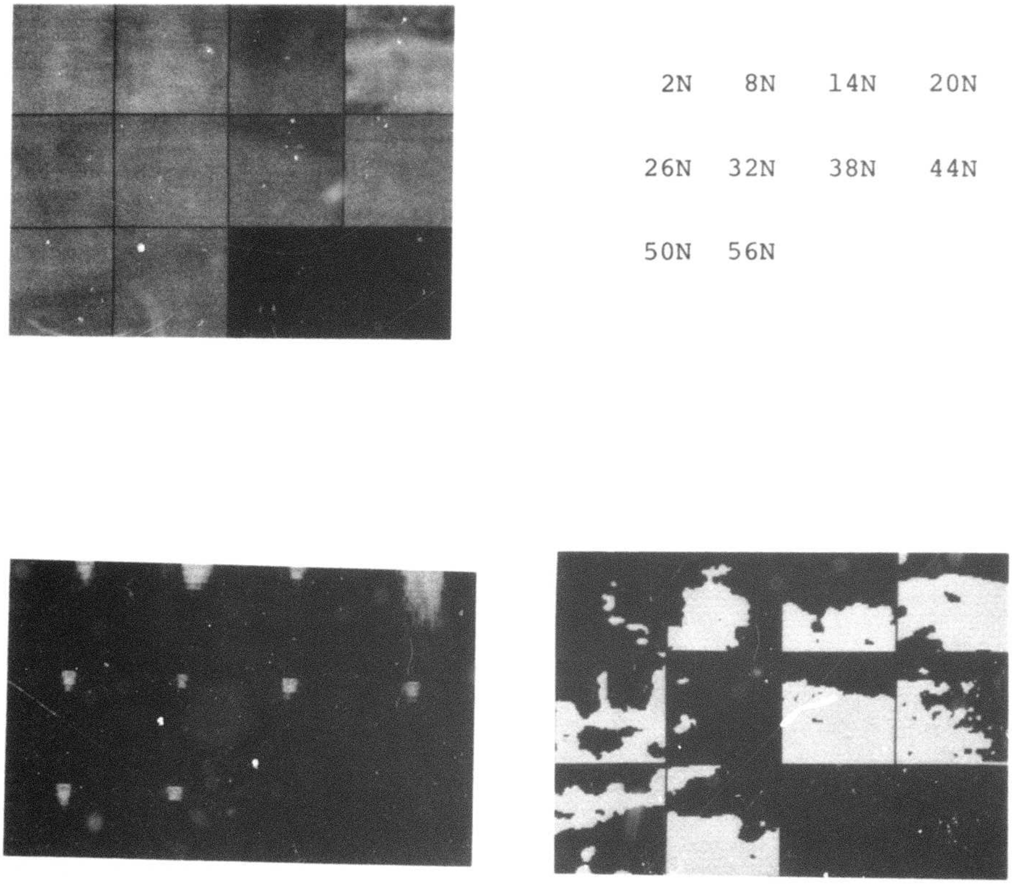
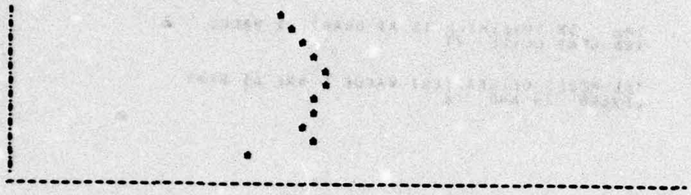


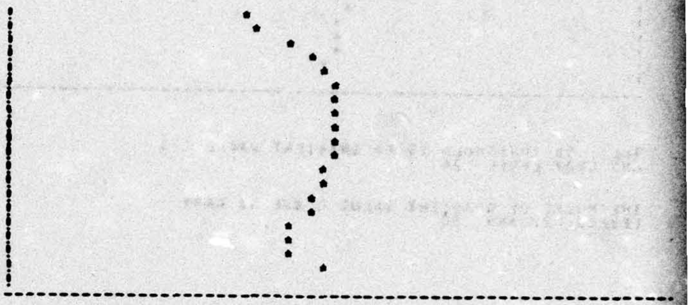
Figure 5.3d. 10 noise windows.

```
 INPUT ELEMENT-VERSION NAME:
              X
NVL2DHISTS   3T
GRADIENT VALUE      AVERAGE GRAY LEVEL      CUMULATIVE PERCENT
      10                    20                      .06
       9                    26                      .46
       8                    25                     1.42
       7                    26                     2.80
       6                    26                     4.65
       5                    27                     8.40
       4                    27                    13.73
       3                    26                    21.55
       2                    25                    37.98
       1                    24                    72.30
       0                    23                   100.00
```



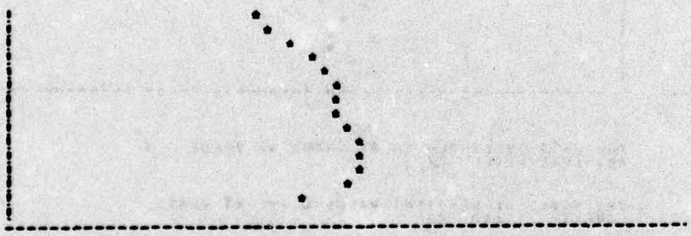
```
THE    5% THRESHOLD IS AT GRADIENT VALUE    5
AND GRAY LEVEL  26

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  24 AND  27
```

```
 INPUT ELEMENT-VERSION NAME:
              X
NVL2DHISTS   6T
GRADIENT VALUE      AVERAGE GRAY LEVEL      CUMULATIVE PERCENT
      18                    27                      .03
      17                    24                      .22
      16                    24                      .36
      15                    24                      .58
      14                    26                      .95
      13                    26                     1.29
      12                    27                     1.60
      11                    27                     2.28
      10                    28                     3.14
       9                    28                     4.31
       8                    28                     5.51
       7                    28                     7.20
       6                    28                     8.68
       5                    28                    10.13
       4                    27                    11.45
       3                    26                    13.73
       2                    24                    22.53
       1                    21                    57.13
       0                    20                   100.00
```



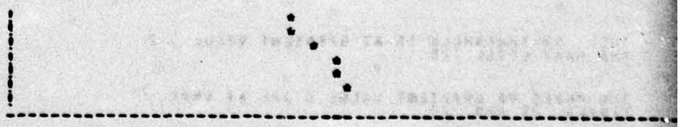
```
THE    5% THRESHOLD IS AT GRADIENT VALUE    8
AND GRAY LEVEL  27

THE MODES OF GRADIENT VALUE ( ARE AT GRAY
LEVELS  16 AND  32
```

```
 INPUT ELEMENT-VERSION NAME:
              X
NVL2DHISTS   8T
GRADIENT VALUE      AVERAGE GRAY LEVEL      CUMULATIVE PERCENT
      13                    25                      .03
      12                    29                      .22
      11                    30                      .55
      10                    30                     1.08
       9                    30                     2.28
       8                    29                     4.25
       7                    28                     5.26
       6                    28                     6.96
       5                    28                     9.48
       4                    27                    11.97
       3                    26                    15.51
       2                    24                    23.21
       1                    22                    59.00
       0                    21                   100.00
```



```
THE    5% THRESHOLD IS AT GRADIENT VALUE    7
AND GRAY LEVEL  28

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  17 AND  34
```

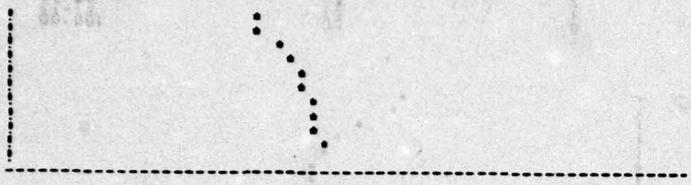
```
 INPUT ELEMENT-VERSION NAME:
              X
NVL2DHISTS   34T
GRADIENT VALUE      AVERAGE GRAY LEVEL      CUMULATIVE PERCENT
       5                    29                      .12
       4                    28                     1.45
       3                    28                     3.69
       2                    26                     7.73
       1                    24                    38.26
       0                    24                   100.00
```



```
THE    5% THRESHOLD IS AT GRADIENT VALUE    2
AND GRAY LEVEL  26

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  23 AND  27
```

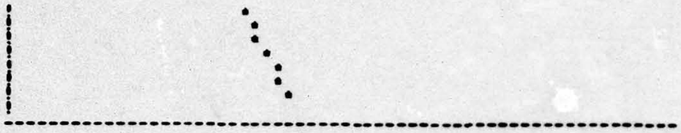
Figure 5.4. Graph of selected threshold as a function of percentage edge value cutoff. Abscissa: gray value increases to the right; ordinate: gradient decreases up the axis.

```
 INPUT ELEMENT-VERSION NAME:
             X
NVL2DHISTS  54T
GRADIENT VALUE       AVERAGE GRAY LEVEL       CUMULATIVE PERCENT
       9                      27                            .15
       8                      26                            .49
       7                      26                            .77
       6                      26                           1.43
       5                      25                           2.49
       4                      25                           3.82
       3                      24                           5.79
       2                      23                          10.99
       1                      21                          39.58
       0                      21                         100.00
```



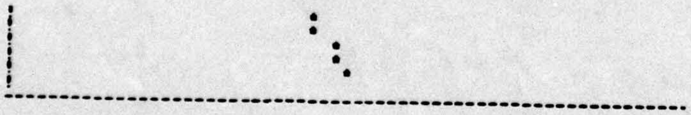
```
THE    5% THRESHOLD IS AT GRADIENT VALUE    3
AND GRAY LEVEL  24

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  22 AND  30
```

```
 INPUT ELEMENT-VERSION NAME:
             X
NVL2DHISTS  58T
GRADIENT VALUE       AVERAGE GRAY LEVEL       CUMULATIVE PERCENT
       6                      24                            .12
       5                      23                            .34
       4                      23                           1.14
       3                      22                           2.76
       2                      21                           8.19
       1                      21                          37.46
       0                      20                         100.00
```



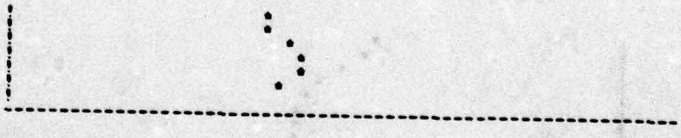
```
THE    5% THRESHOLD IS AT GRADIENT VALUE    2
AND GRAY LEVEL  21

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  20 AND  22
```

```
 INPUT ELEMENT-VERSION NAME:
             X
NVL2DHISTS  51R
GRADIENT VALUE       AVERAGE GRAY LEVEL       CUMULATIVE PERCENT
       4                      29                            .74
       3                      28                           2.43
       2                      28                           5.23
       1                      26                          28.53
       0                      26                         100.00
```



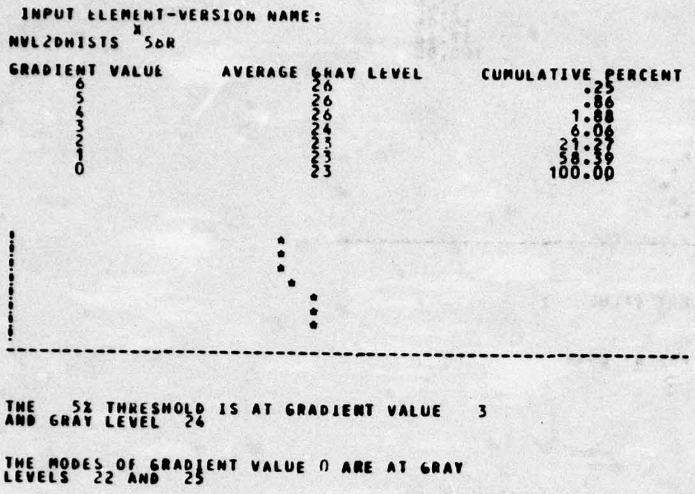
```
THE    5% THRESHOLD IS AT GRADIENT VALUE    2
AND GRAY LEVEL  28

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  25 AND  29
```

```
 INPUT ELEMENT-VERSION NAME:
             X
NVL2DHISTS  52R
GRADIENT VALUE       AVERAGE GRAY LEVEL       CUMULATIVE PERCENT
       5                      23                            .03
       4                      25                            .58
       3                      25                           1.54
       2                      24                           3.16
       1                      22                          23.64
       0                      22                         100.00
```

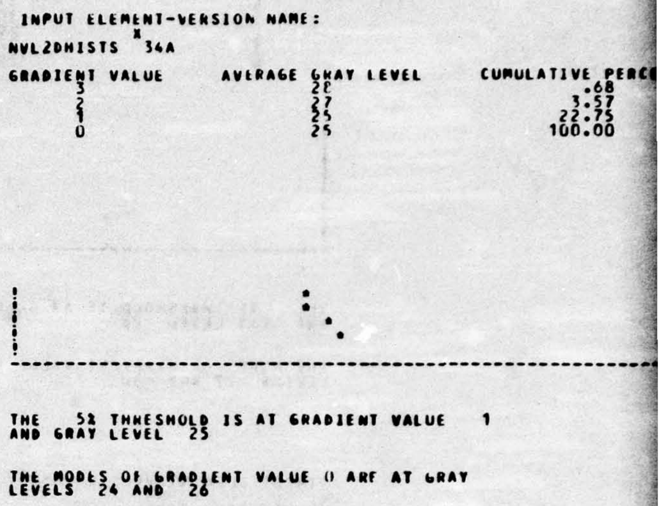


```
THE    5% THRESHOLD IS AT GRADIENT VALUE    1
AND GRAY LEVEL  22

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  21 AND  23
```

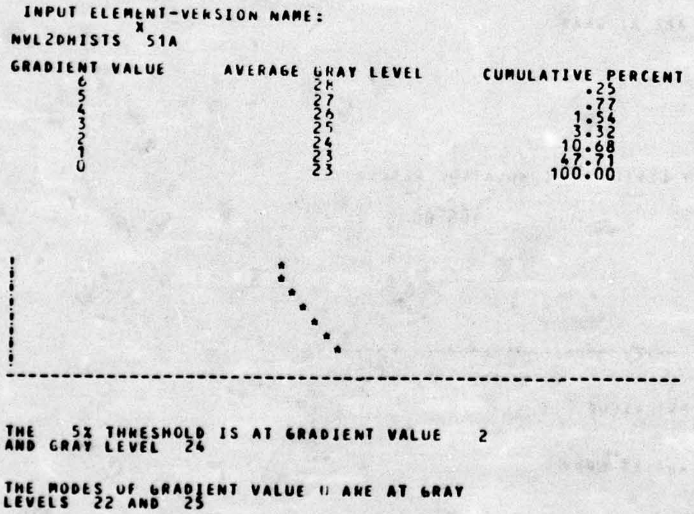
Figure 5.4 (continued)



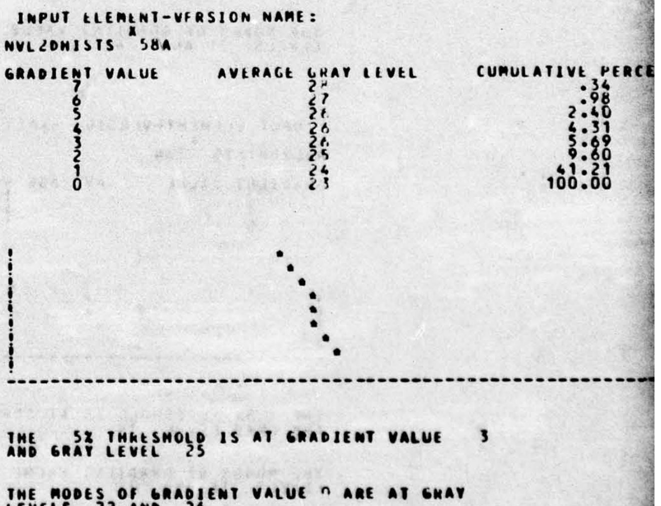
```
 INPUT ELEMENT-VERSION NAME:
              X
NVL2DHISTS   56R

GRADIENT VALUE       AVERAGE GRAY LEVEL       CUMULATIVE PERCENT
      6                     26                           .25
      5                     26                           .86
      4                     26                          1.88
      3                     24                          6.06
      2                     23                         21.27
      1                     23                         58.39
      0                     23                        100.00

THE    5% THRESHOLD IS AT GRADIENT VALUE    3
AND GRAY LEVEL  24

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  22 AND  25
```



```
 INPUT ELEMENT-VERSION NAME:
              X
NVL2DHISTS   34A

GRADIENT VALUE       AVERAGE GRAY LEVEL       CUMULATIVE PERC
      3                     28                           .68
      2                     27                          3.57
      1                     25                         22.75
      0                     25                        100.00

THE    5% THRESHOLD IS AT GRADIENT VALUE    1
AND GRAY LEVEL  25

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  24 AND  26
```



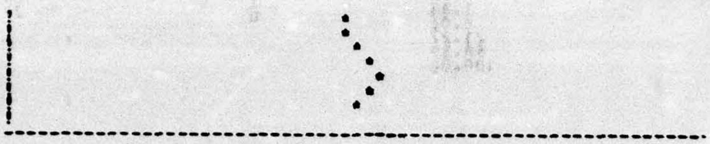
```
 INPUT ELEMENT-VERSION NAME:
              X
NVL2DHISTS   51A

GRADIENT VALUE       AVERAGE GRAY LEVEL       CUMULATIVE PERCENT
      6                     28                           .25
      5                     27                           .77
      4                     26                          1.54
      3                     25                          3.32
      2                     24                         10.68
      1                     23                         47.71
      0                     23                        100.00

THE    5% THRESHOLD IS AT GRADIENT VALUE    2
AND GRAY LEVEL  24

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  22 AND  25
```



```
 INPUT ELEMENT-VERSION NAME:
              X
NVL2DHISTS   58A

GRADIENT VALUE       AVERAGE GRAY LEVEL       CUMULATIVE PERC
      7                     28                           .34
      6                     27                           .98
      5                     26                          2.40
      4                     26                          4.31
      3                     26                          5.69
      2                     25                          9.60
      1                     24                         41.21
      0                     23                        100.00

THE    5% THRESHOLD IS AT GRADIENT VALUE    3
AND GRAY LEVEL  25

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  22 AND  26
```

Figure 5.4 (continued)

```
 INPUT ELEMENT-VERSION NAME:
        X
NVL2DHISTS  2N

GRADIENT VALUE       AVERAGE GRAY LEVEL      CUMULATIVE PERCENT
      6                      29                          .03
      5                      30                          .18
      4                      31                          .95
      3                      30                         3.57
      2                      29                        15.05
      1                      28                        57.86
      0                      28                       100.00
```

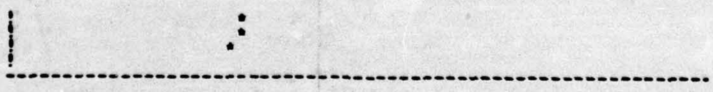


```
THE   5% THRESHOLD IS AT GRADIENT VALUE    2
AND GRAY LEVEL  28

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  27 AND  30
```

```
 INPUT ELEMENT-VERSION NAME:
        X
NVL2DHISTS  26N

GRADIENT VALUE       AVERAGE GRAY LEVEL      CUMULATIVE PERCENT
      3                      23                          .06
      2                      22                         4.56
      1                      22                        46.54
      0                      22                       100.00
```

```
THE   5% THRESHOLD IS AT GRADIENT VALUE    1
AND GRAY LEVEL  22

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  21 AND  24
```

```
 INPUT ELEMENT-VERSION NAME:
        X
NVL2DHISTS  38N

GRADIENT VALUE       AVERAGE GRAY LEVEL      CUMULATIVE PERCENT
      2                      18                         2.25
      1                      19                        30.62
      0                      19                       100.00
```



```
THE   5% THRESHOLD IS AT GRADIENT VALUE    1
AND GRAY LEVEL  19

THE MODES OF GRADIENT VALUE 0 ARE AT GRAY
LEVELS  16 AND  21
```

Figure 5.4 (continued)

5.3 Non-Maximum Suppression of Edge Responses

The approach used in Section 5.1 was to enrich the population of points having high edge values by using an operator which responded maximally to ramps. In this section, we look into a complementary approach, that of screening the high edge value population by limiting membership to "best" exemplars of high edge value.

Local non-maximum suppression of edge points was mentioned in passing in Section 4.2. It corresponds to the simultaneous application at each image point of the following rule:

> Determine the maximum value of all points lying in a predetermined neighborhood template about (but excluding) the given point. If the computed maximum exceeds the value at the given point, replace the given point value by zero.

The particular neighborhood selected depends on the choice of edge detectors. Obviously, one wants a single maximal response across the edge ramp. Thus suppression should take place across the edge but not along it. Thus an edge detector for vertical edges would require suppression in the horizontal direction. As a corollary, the chosen edge operator should be one which exhibits its peak at the midpoint of the ramp. This rules out the median difference edge detector of Section 5.1. For the 4x4 mean difference operator (for vertical edges), the following horizontal suppression template has been used:

```
x x       x x
x x x o x x x
x x       x x
```

where o is the given point and x's represent neighbors whose values are interrogated. A similar template may be used for vertical suppression. Figure 5.5 illustrates thinned edge response based on local non-maximum suppression.

Several by-products of this transformation are of interest. First, interiors of regions of homogeneous edge value survive this process. Thus smooth gradients and thick edges are not necessarily thinned. It is not sufficient to change the rule to "If the computed maximum is not less than..." This would eliminate entire regions of homogeneous response but it would also most likely eliminate a sizable fraction of true object edge which happens to have homogeneous response. An effective solution would be to use not the difference of means over 4x4 regions but the difference of sums over 4x4 regions. This would eliminate the truncation error which causes so many edge values to appear identical. The likelihood of equal adjacent untruncated values is small. After non-maximum suppression the resulting thinned vlaues could be divided by the neighborhood size to produce gradient values. Another way to achieve much of the same result is to add random noise of fractional amplitude to the edge picture before suppression. The resultant real-valued image is unlikely to have

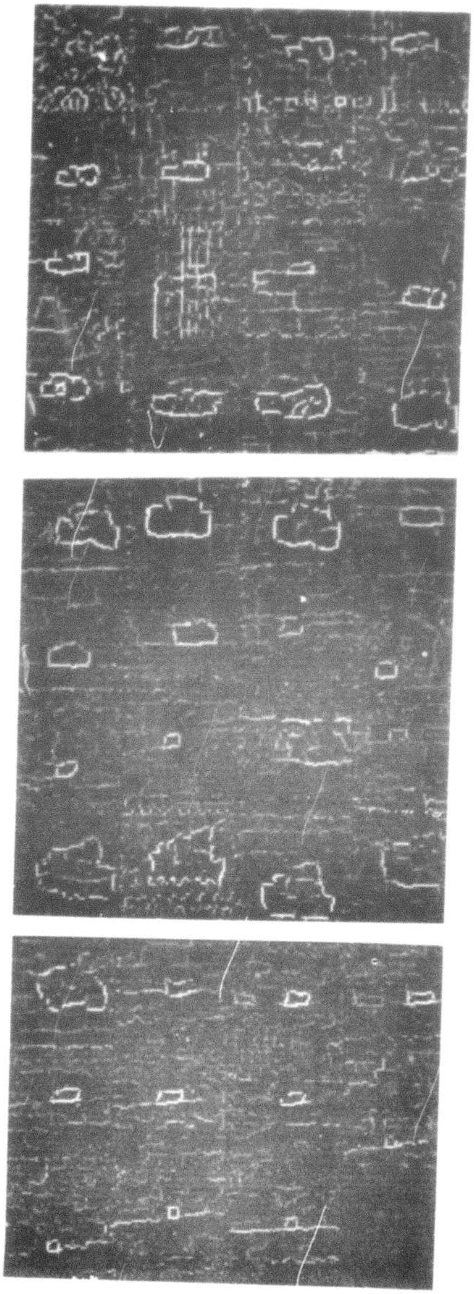


Figure 5.5. Local non-maximum suppression of edge detector responses, for the same set of images as in Figure 5.3.

a. 43 tanks.

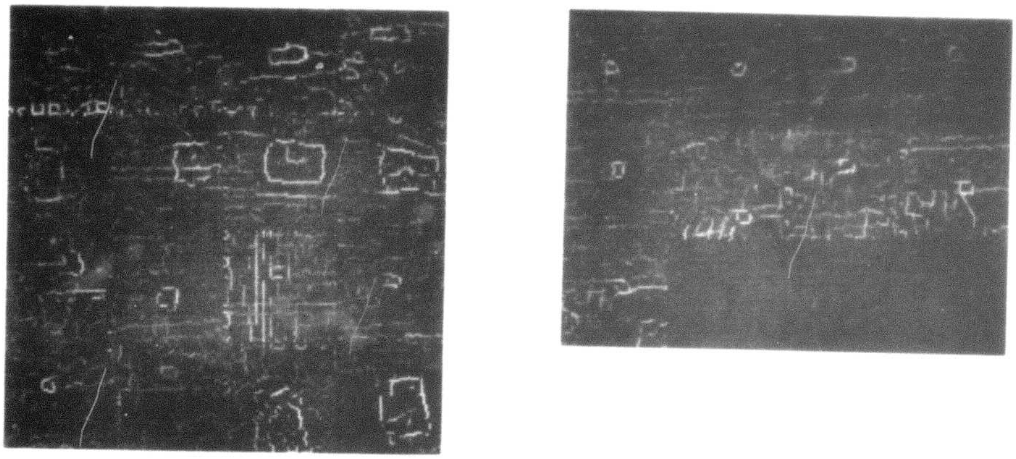


Figure 5.5b. 25 trucks

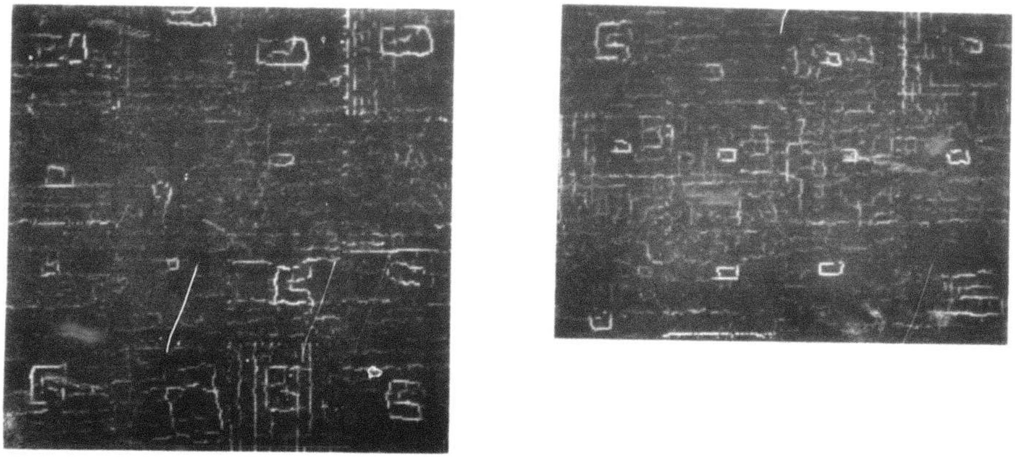


Figure 5.5c. 28 APCs

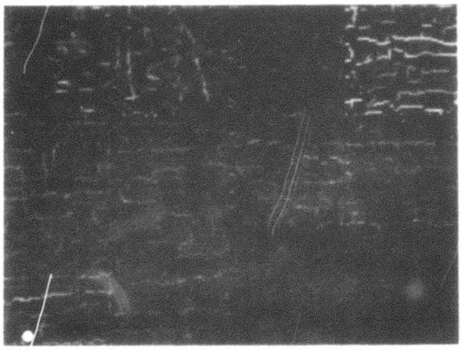


Figure 5.5d. 10 noise windows.

homogeneous regions.

A second by-product of this process is the continued presence of noise. Indeed, homogeneous regions of low gradient value are much more probable than similar regions of higher gradient values. One might consider that the signal-to-noise ratio has decreased; however, the tie-breaker methods of the previous paragraph should keep the signal-to-noise ratio fairly constant. Thus there is no reason to believe that the thinned response is a better sample for threshold selection. It should be no worse, though, and it does have the property that its expected position is along the midline of the edge ramp. This could simplify the modelling aspect since there is now a closer relationship between object size and thinned edge response. The amount of thinned edge (above some noise-elimination threshold) should be proportional to the square of the object area. Given the minimal detectable target size for a particular situation, one can eliminate windows with insufficient edge contribution.

There are a number of additional uses for local maxima edge values. Section 6.2 discusses the coincidence of local edge maxima and borders of connected components. We discuss here an application to automatic thresholding of images containing objects from two or more gray scale populations. Consider the two-class case, referring to two gray level distributions for two object populations (O_1 and O_2) and a third density for the background. Assuming nonoverlapping contours and isotropic edge response, the different edge populations should cluster in isolated regions of the 2-D histograms. The clusters would tend to blend if local non-maximum suppression were not used. Since edge-population increases with the square root of gray scale population, the cluster sizes should be less sensitive to object size. Inasmuch as the current NVL data base does not satisfy the multipopulation assumption, this proposal has not been tested.

6. Post-Processing and Connected Component Extraction

Thresholding produces a binary image. Not everthing in that binary image corresponds to an object, however. Small gray level variations in the background can produce point clusters which are above threshold. Also, the border of an object can have thin prominences which (perhaps) correspond to non-uniformities along its edge (e.g., heat plumes). This section discusses two approaches which eliminate such spurious points. The first, using shrinks and expands, was introduced in the first quarterly report; it is further analyzed here. The second proposes a new method of noise component rejection based on the heuristic that the borders of true objects should correspond to high edge response, while borders of noise regions should not.

6.1 Further Experiments with Shrink/Expand

The purpose of a shrink/expand process is to delete regions of small size or low density (many holes). Compact regions of moderate size should survive this process. A shrink operation deletes (makes 0) any point whose neighborhood contains more than a given number of 0's. This affects isolated points as well as certain boundary points. A succession of shrinks thus may eliminate small isolated regions or prominences. It may also tend to reduce the size of actual object regions. To restore such reduced regions to a more correct size, a succession of expand operations is performed. An expand operation makes any point 1 whose neighborhood contains more than a given number of 1's. The number of expands should equal the number of shrinks and should be chosen according to the maximum size of compact region whose deletion is desired. In this case, it was desirable to delete any region of nine or fewer points in the sampled windows. This called for two shrinks followed by two expands.

The number of neighbors, k, required in order to shrink or expand a point was investigated previously and was set to 3. It has been noticed that this causes certain anomalies. While a shrink operation with k=3 is guaranteed to delete border points of an object, an expand operation with k=3 does not necessarily add any new points. In particular, a 5x5 block will shrink down to a single point after two shrinks but will not expand thereafter. If we decide instead to expand with k=1, undesirable growth occurs (Figure 6.1). In practice, this anomaly did not affect the appearance of objects in thresholded windows. However, it may be necessary in the future to revert to k=1 for both shrinks and expands, if the shrink/expand process is still found to be necessary.

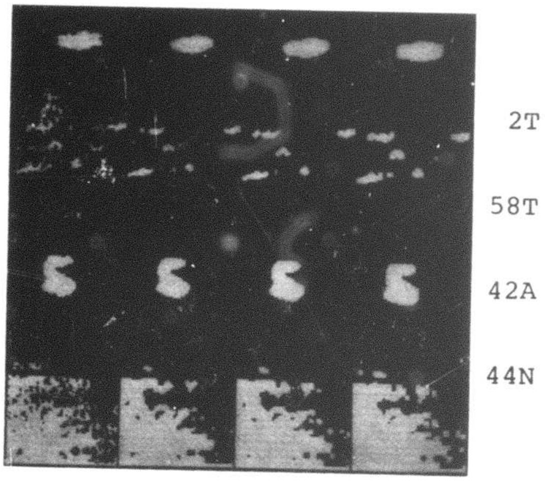


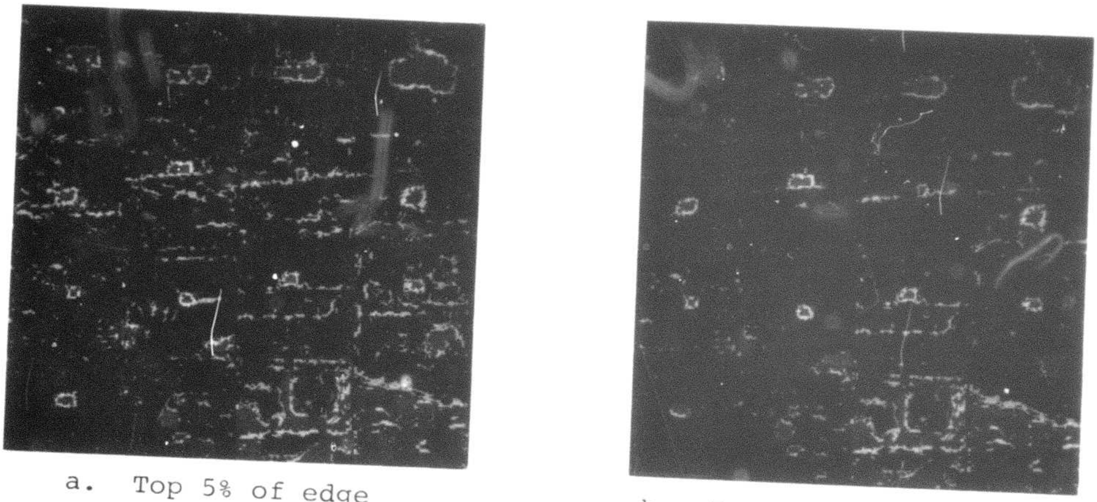
Figure 6.1. Effect of varying k in the expand process; k = 3 for the shrink process.

Column 1: original thresholded windows.
Columns 2-4: shrink with k = 3; expand with k = 3, 2, 1, respectively.

6.2 Noise Component Rejection Based on Lack of Edge Evidence

Several criteria are relevant to how a set of points defines an object in a FLIR scene. First, the gray levels in the set should be approximately the same. Second, they should be distinguishable from the background (at least from that part of it that lies just outside the border of the set). Third, the border points of the set should correspond closely to the strongest edges in the vicinity of the set. Furthermore, these strongest edges should surround the set. Finally, the points themselves should form a fairly compact region (few holes or thin prominences or deep indentations) and should be isolated from any regions of similar description (i.e., little interpenetration or adjacency). Clearly, one could add subtler distinctions involving known shapes or context. Nor is there any contention that all the possible criteria have been included. Nonetheless, the cooccurrence of features derived from the above criteria should be strong evidence that the component is not a noise region. This section describes two experiments using border points as sources of evidence.

Section 5.3 discussed a method of extracting thinned edges from a scene. This procedure was carried out in a modified form to produce Figure 6.2. In this figure, the thinned edge response was thresholded to preserve only the top 5% of the edge values. The resultant mask was composed (ORed) with the thresholded and post-processed (binary) windows. One can see that there is a large degree of corres-



a. Top 5% of edge responses overlaid on thresholded windows.

b. Top 3% of edge responses overlaid on thresholded windows.

Image reference numbers: 3T 6T 8T 24T

34T 54T 58T 31T

52R 58R 34A 54A

58A 2N 26N 38N

Figure 6.2. Correspondence of high edge responses to object boundaries.

pondence between the borders of the regions and the thinned edges. Note that the edge points do not necessarily surround the object regions completely nor do noise regions lack edge response. However, the degree of closure does differ markedly in most cases and should aid in distinguishing objects from noise regions. In general, the larger the region, the greater the degree of closure to be demanded for classification as an object. It is planned to develop a statistical model for this phenomenon.

A second experiment was designed to make use of the heuristic that object regions should be surrounded by the strongest edges in the scene. The ideal 2-D histogram (Figure 6.3) of a scene containing an object and background with noise provides the motivation for this analysis. Points at A represent the background while B represents object points and some background noise. Obviously, we cannot separate object points from background noise points on the basis of gray level alone. The base of the U-shaped region contains high-value edge points. If a point lies within an object region it should be the case that any path from it to the background region must pass through the high value edge region, since we assume that the high value edges surround the object region. However, this should not hold for noise values in the background. Referring back to Figure 6.3, this means that paths from object points to background points should correspond to trips around the U whereas paths from noise background points to the background avoid the base of

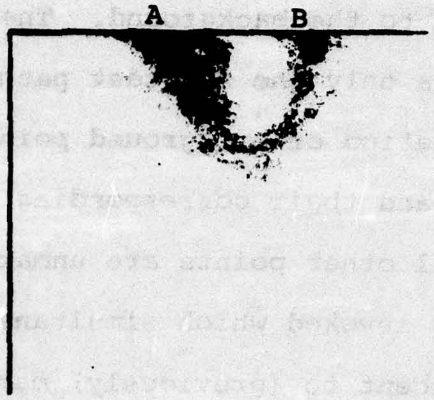
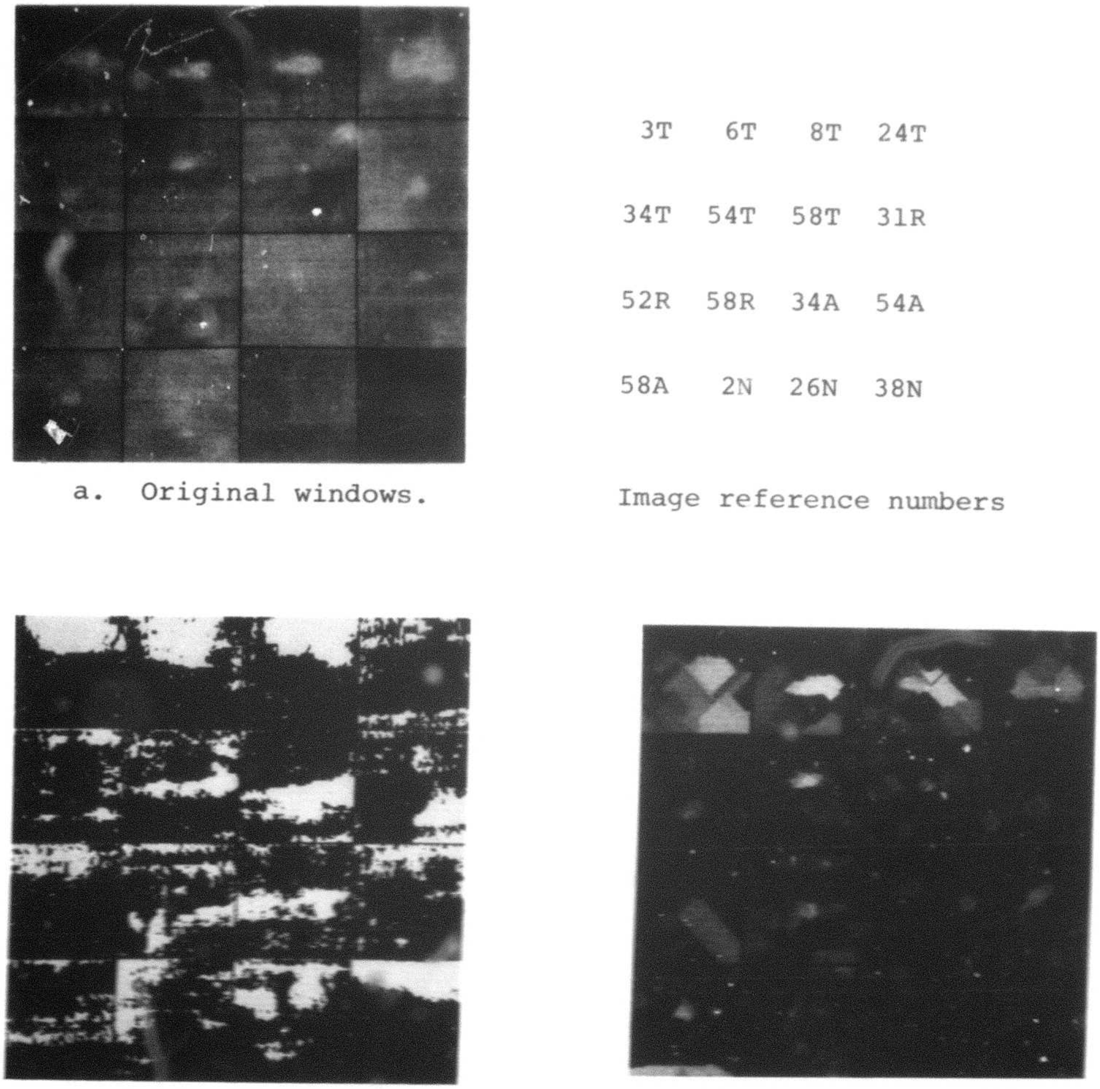


Figure 6.3. Ideal 2-D histogram of a scene containing an object and background with noise.

the U (take the short cut across the U).

A program has been written to test this hypothesis. Computationally, it is impossible to consider all paths from an object point to the background. The method we have implemented considers only the shortest path. It operates as follows: A population of background points is initially identified (marked) and their corresponding thinned edge values assigned. All other points are unmarked. An iterated parallel step is now invoked which simultaneously marks all unmarked points adjacent to (previously) marked points. Each marked point receives a value corresponding to the maximum of its edge value and the maximum edge value among its marked neighbors. It should be clear that any point marked on the kth iteration is at distance k from the initial background region. Moreover, the value assigned to it is the maximum edge value along this shortest path to the background region. Thus object regions enclosed by high edges should be colored in with the high edge values.

Figure 6.4 shows the result of applying this algorithm to a number of windows. It is apparent that the choice of initial background region is crucial. The basic criterion for the initial point set is that it be well distributed throughout the background. In practice, the low 25% of the gray levels were chosen. A better selection might have resulted if only low gray level points with zero gradient value had been employed. In addition, one might wish to discard all isolated points and to include the border points



3T	6T	8T	24T
34T	54T	58T	31R
52R	58R	34A	54A
58A	2N	26N	38N

a. Original windows.

Image reference numbers

b. Initial sets for propagation.

c. Regions as painted by propagation process.

Figure 6.4. Region painting based on propagation of maximum edge responses.

of the window (this relates to certain propagation anomalies which produced extended regions).

The results of this experiment are encouraging. Object regions end up being fairly well defined without strong reliance on gray level. The strongly parallel nature of the algorithm guarantees that no more than N iterations are required to color in an nxn window. With proper choice of an initial set, the expected number of iterations is just the radius of the object.

7. Feature Selection and Target Classification

7.1 General Approach

Once localized bright areas ("objects") have been detected and isolated from the input scene, the problem of identifying and classifying those of particular interest can, in a first approximation, be divided into four steps. First, one determines what kind of features are practical to obtain from the images available and might be useful in discrimination. Next, the gross structure of the classification scheme to be used can be sketched. Third, for each decision point in the classification procedure, some subset of the available features should be identified as being of relevance. Finally, the detailed decision procedures at each node can be determined by "training" on samples of identified targets, and by estimating the accuracy of the resulting classification structure.

While in practice these steps are not entirely independent, our classifier development could be described as several iterations of this basic sequence. The current state of development is probably best described by going through a delineation of the choices made in the most recent cycle.

7.2 Feature Selection

Initial feature selection, though important to the success of the overall classification procedure, is closely tied to the techniques by which potential targets are to be selected prior to classification, and will only be sketched here.

Those variables currently used in classification fall into three groups: size-dependent variables such as area, height, width and perimeter; (nearly) size-independent features such as ratios of size features and target moments of inertia; and brightness features, including the difference between the average gray level of the object and its perimeter, the standard deviation of the brightness over the object, and other features obtainable from the gray level histogram of the object. Features are chosen to reflect apparent differences between "target" objects and background objects <u>and</u> to be independent of such irrelevant factors as shift in overall brightness of a scene, or a change in orientation or scale.

A further, more technical, basis for selecting input features is that they be "appropriate" to the types of classifiers actually implemented in the decision structure. Measures which yield more-or-less normally distributed, more-or-less independent features are likely to be more readily usable than features chosen without regard for their units and dimensions. (One approach to the feature selection problem is to include all conceivable forms of every variable which seems to be of interest. Only when virtually infinite files of preclassified data are available can this "maximum ignorance" approach be effective.)

7.3 Classification Scheme

Constraints on the classification structure to be employed are that it be robust, powerful, fast, and intelligible. (As usual, the optimization criteria conflict somewhat.) All of these criteria (except, perhaps, speed) suggest a hierarchical decision structure, such as that sketched in Fig. 7.1. An optimum structure will obviously represent a compromise between a very shallow tree (for speed), a deep and heavily foliated structure (for high accuracy), and a "simple" structure (for intelligibility and plausibility).

Several features of the structure shown seem now to be fixed. All data is "prescreened" on the basis of one or two "powerful" variables (here: the difference in object and perimeter gray level should be positive; the area should be at least 20 pixels.) Possible objects are then sorted on the basis of size into small objects, for which the only further decision is whether the object is target or background, and large objects, whose identification can be more accurately determined. The detailed structure of this latter determination is not as firmly established as the other parts of the decision structure. That it is a reasonable structure is suggested by the relative ease in distinguishing the various target types with our present feature set.

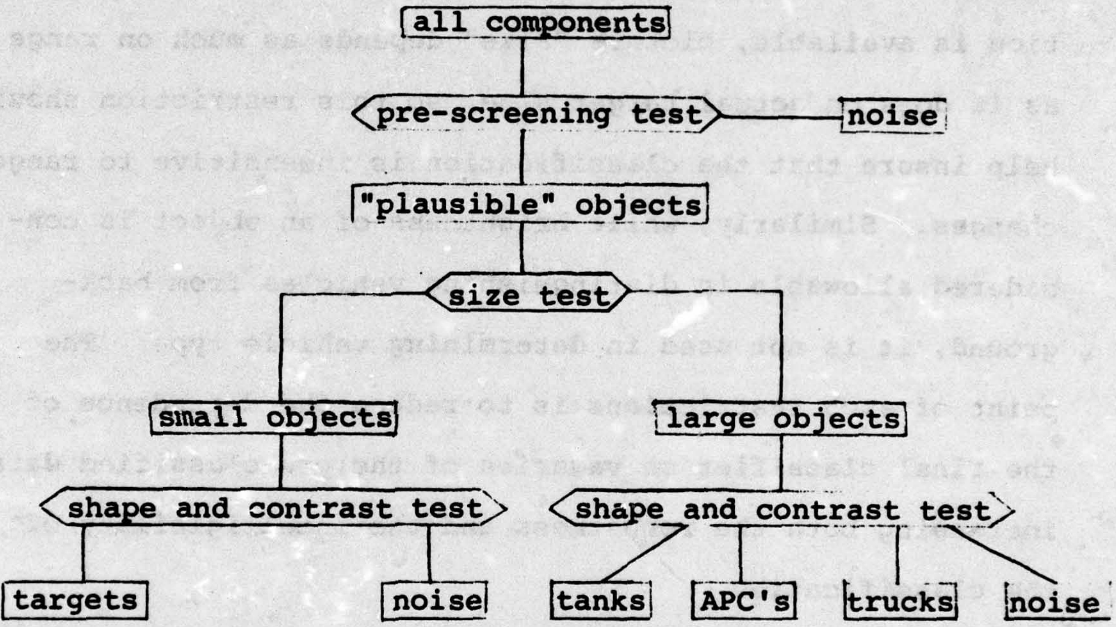


Figure 7.1. A hierarchical decision structure for target classification.

7.4 Feature Reduction

Selection of the set of features actually used at each node is done in two stages. First, features are restricted to those "logically allowable" at the given node. For example, after the decision as to whether an object is large enough for detailed examination, only "shape" and "gray level" variables are used. Since no absolute size information is available, picture "size" depends as much on range as it does on actual target size, so this restriction should help insure that the classification is insensitive to range changes. Similarly, while brightness of an object is considered allowable in distinguishing vehicles from background, it is not used in determining vehicle type. The point of such restrictions is to reduce the dependence of the final classifier on vagaries of the pre-classified data, increasing both the robustness and the intelligibility of the classification.

After this "logical" preselection, a further reduction in the number of features at each decision point can be made on statistical grounds. Standard statistical techniques (analysis of covariance, multiple discriminant analysis) combined with empirical comparisons of feature subsets are used to select a relatively small number of effective features at each decision point. Again, the purpose is to increase the stability of the classifier without decreasing its accuracy. In addition, features which are not important at <u>any</u> node can be identified and eliminated from the scheme entirely.

7.5 Classifier Selection, Tuning and Testing

Once a plausible set of features, a decision tree, and a distribution of the features among the decision nodes has been chosen, the type of discriminant to be used at each node is selected and optimized, and the accuracy of the complete decision tree tested by classification of pre-classified "training" and "test" samples. Since the whole target determination and classification system is still in flux, no attempt to make realistic estimates of the error rates so far obtained has been made. The results of some trial classifiers on a training set are described below, however. In all cases, the training set consisted of 492 putative objects, obtained as the connected components of 91 thresholded scenes containing 87 identified targets.

In part because of the availability of tested programs for optimizing linear and quadratic discriminant classifiers, and in part because of the general claim of robustness for Fisher (linear) classification, all binary classification decisions can be based on the Fisher discriminant with an appropriately selected *a priori* branching ratio. (The adequacy of such a procedure is dependent on the choice of appropriate features, as mentioned above.) Comparison of such linear discriminants with quadratic discriminants on the same features was also carried out, a typical case being shown in Table 7.1. The decision tree of Fig. 7.1 includes one three-way decision, for which various possibilities can be used directly (e.g., a "voting" or a "one-against-the-

	classified as	
	truck	noise
truck	10	7
noise	1	35

a) Fisher linear discriminant.

	classified as	
	truck	noise
truck	15	2
noise	5	31

b) Quadratic discriminant.

Table 7.1. A comparison of a linear discriminant with a quadratic discriminant.

rest" scheme with any binary classifier, or a maximum-likelihood classifier), or the multiple decision may be replaced by a pair of binary decisions.

Table 7.2 shows the confusion matrix for the above decision tree using Fisher discriminant classifiers at all binary nodes and a quadratic maximum-likelihood classifier at the ternary node. While each node decision is optimized for this classification, only limited tuning of the full system was done. Notice that while 38 errors were made (36 on the 171 objects which passed the pre-selector), only 8 targets of 87 were identified as background and only 17 "noise" objects were proposed as targets, the remaining errors being presumably less crucial mislabelings of true targets. Still, it is amply clear that further development is required, especially for the "large object" discrimination of the tree, representing all those branches emanating from the LARGE node.

	classified as	
	object	noise
object	85	2
noise	86	319

a) Result of pre-screening.

	classified as	
	target	noise
target	28	2
noise	3	49

b) Result of small object classification.

	classified as			
	tank	APC	truck	noise
tank	24	0	2	2
APC	3	8	2	1
truck	3	3	7	1
noise	5	6	3	19

c) Result of large object classification.

Table 7.2. Confusion matrices for the hierarchical decision structure.

8. Data Bases

The University of Maryland has acquired three data bases which will be studied and utilized under this contract. The acquisition of more data bases is desirable not only for algorithm testing but also to better represent the variety of natural scene types in which target cueing is needed. The development of a robust model requires a deep understanding of sensor characteristics, tactical environments and target characteristics. What seem to be reasonable heuristics based on one scene type may fail to hold for other types. It remains a continuing shared responsibility of the contractor and the funding agency to widen the test base by acquiring and studying new data.

8.1 The NVL Data Base

This set of images provided by NVL was described in the first quarterly report. Experiments reported there and in this report demonstrate that despite the noisy quality of the images it is possible to extract reliable object components for feature analysis. Of course, images severely degraded by hum or ringing are a noteworthy exception.

In this quarter, all remaining identifiable targets were windowed, median filtered, and thresholded. This finishes the preparation of the complete data base. It has been divided into two subsets, arbitrarily. The first corresponds to 90 windows from tapes A-I. The second consists of all remaining images from tapes J-O (82 windows). Splitting the data base in this way should allow a train and test classification paradigm in the future.

8.2 The Alabama Data Base

This second data base (ALA) is from a non-real-time sensor and is of higher quality than the NVL data base. Far more object detail and far less noise is evident. These images have been read in, grayscale mapped into the range 0-63, and windowed. These 54 target images (Figure 8.1) have been median filtered and should provide an independent test of the threshold selection scheme. Note that the median filtering can change point type noise into a false contour (Fig.8.2). This is the result of large gaps in the histograms produced by quantization remapping. Smoothing (mean filtering) or median filtering over even-size neighborhoods would reduce this effect.

Tape A

Reference No.	Target(s)	Aspect(s)
1	T	S
2	A·J·T	S·3R·S
3	A·T	3R·3F
4	J·T	S·S
5	T	3F
6	T·A·J	S·S·S
7	T	3R
8	T·A	F·S
9	J·T·A	S·S·S
10	T	3F
11	T	3R
12	T·A	3R·3F
13	J·A·T	S·F·F
14	T·A·J	S·S·S
15	T	3F
16	A·T	S·S
17	A·T·J	S·S·3R
18	T	3R
19	T·A	3F·S
20	T·J	R·3R
21	A	R
22	T·A·J	3R·3F·S
23	T	S
24	T	F
25	T	S
26	T	3R
27	T·A	F·F
28	T·A	S·S
29	T·A	R·R
30	T·A·J	S·S·S

Figure 8.1a. Alabama data base. Ground truth.

Tape A

Reference No.	Target(s)	Aspect(s)
31	T·A·J	S·S·S
32	J·A·T	S·F·F
33	B·A·T	S·F·F
34	A·J	3F·S
35	J·T·A	S·S·S
36	T·A·J	S·S·S
37	T·A	3R·3R
38	T·A	3R·3R
39	T·A	3R·3R
40	T·A	3R·S
41	T·A	3R·3R
42	T·A	3R·3F
43	T·A	3F·3R

Legend: A = APC, B = bus, J = jeep,
P = person, T = tank,
S = side, F = front,
R = rear,
3 = 3/4 view

Figure 8.1a (continued)

Tape B

Reference No.	Target(s)	Aspect(s)
44	T	3R
45	T·T	S·S
46	T	S
47	T	S
48	A	3R
49	P·P·P·A	S
50	T	3R
51	T	S
52	T	3R
53	T	F
54	A	3R

Figure 8.1a (continued)

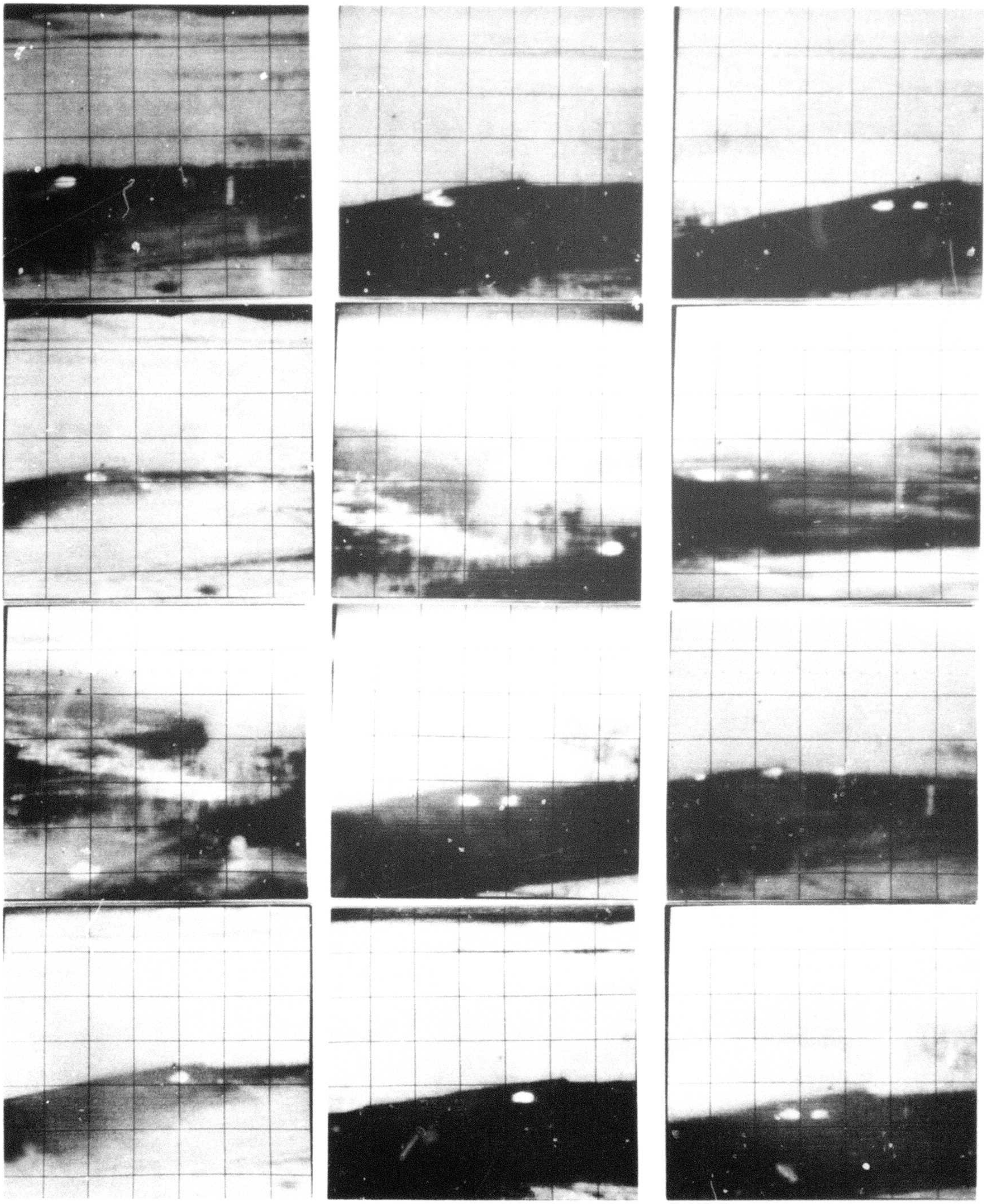


Figure 8.1b. Alabama data base. Images 1-12.

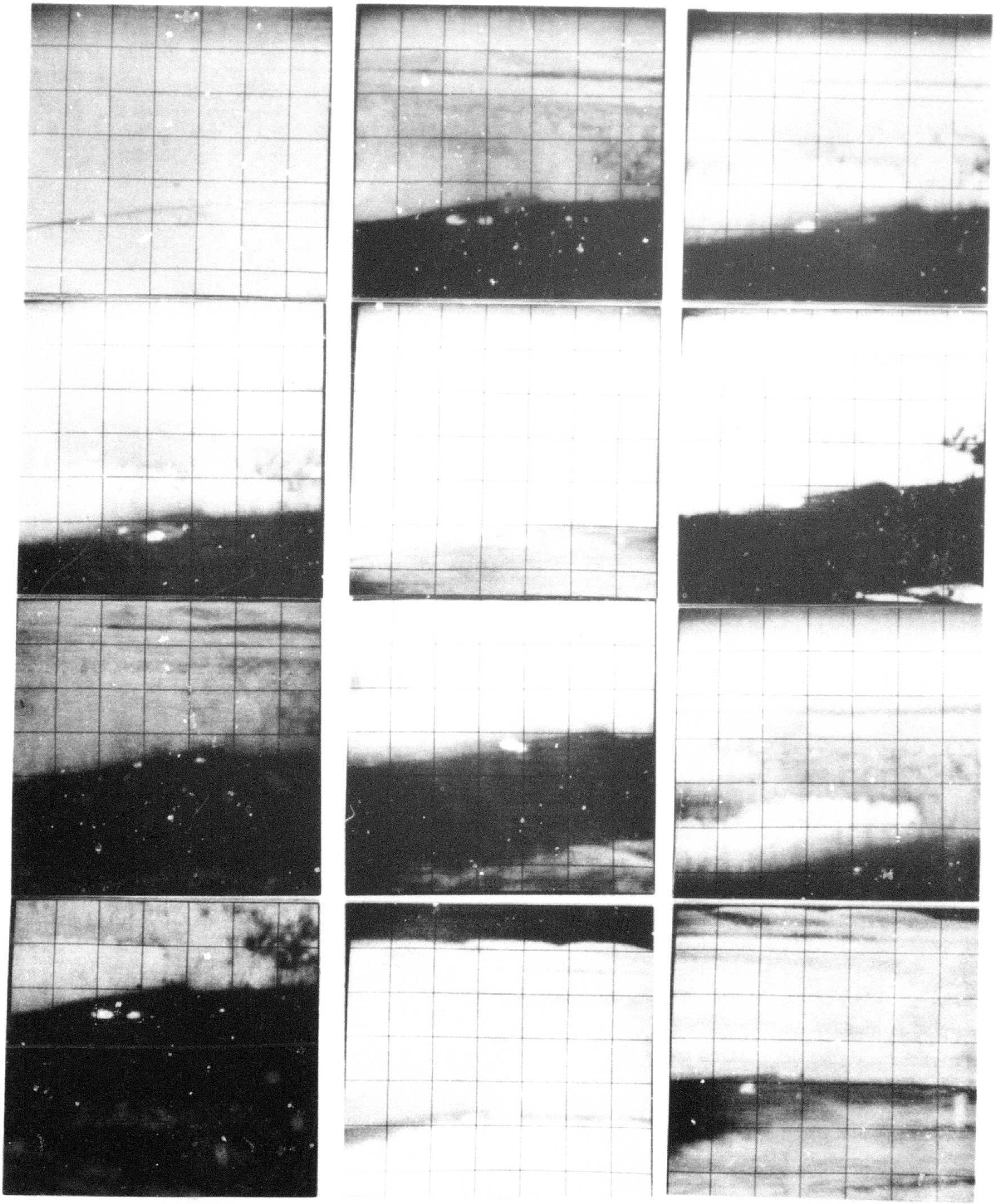


Figure 8.1b (continued). Images 13-24.

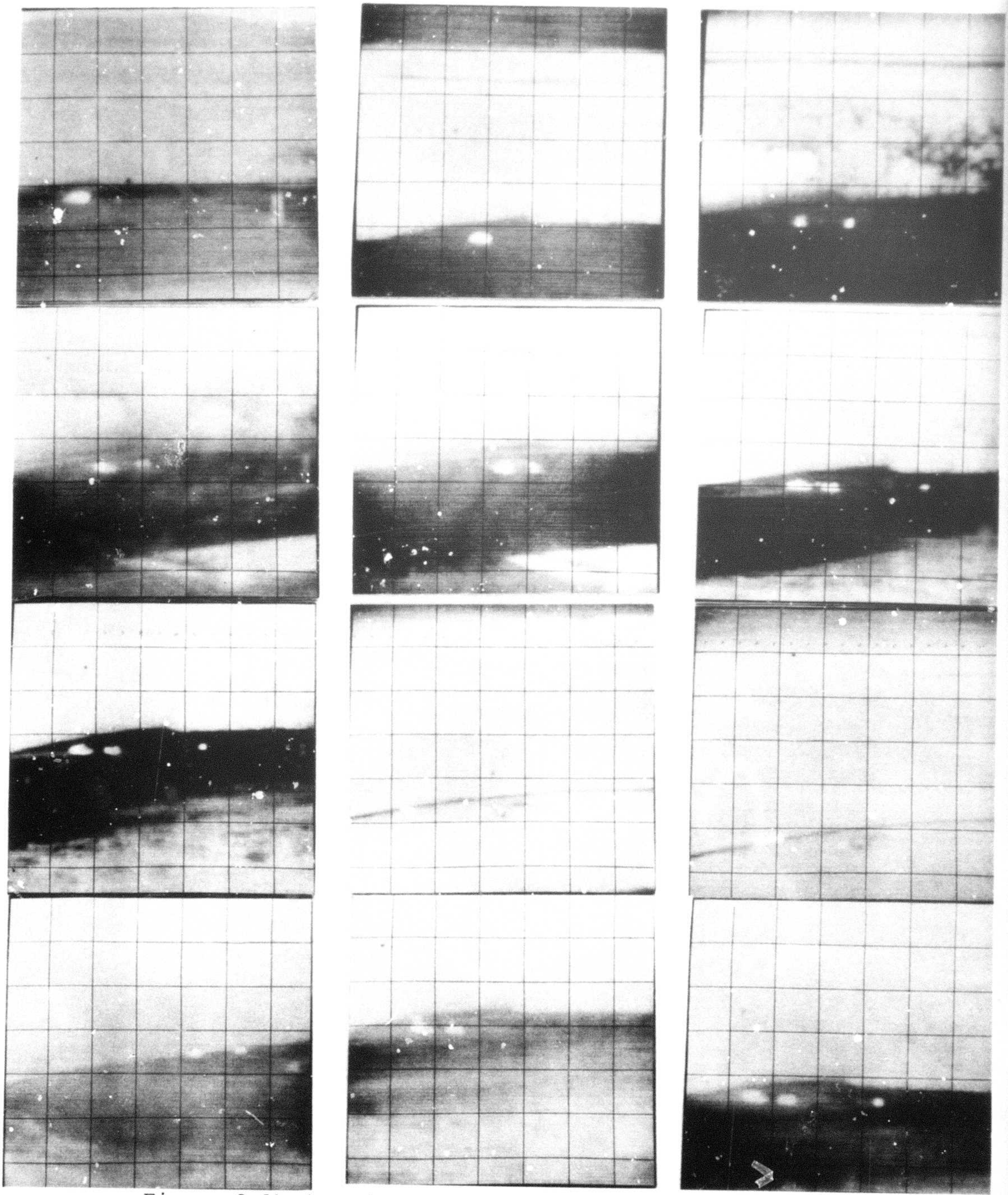


Figure 8.1b (continued). Images 25-36.

Image 37
not available

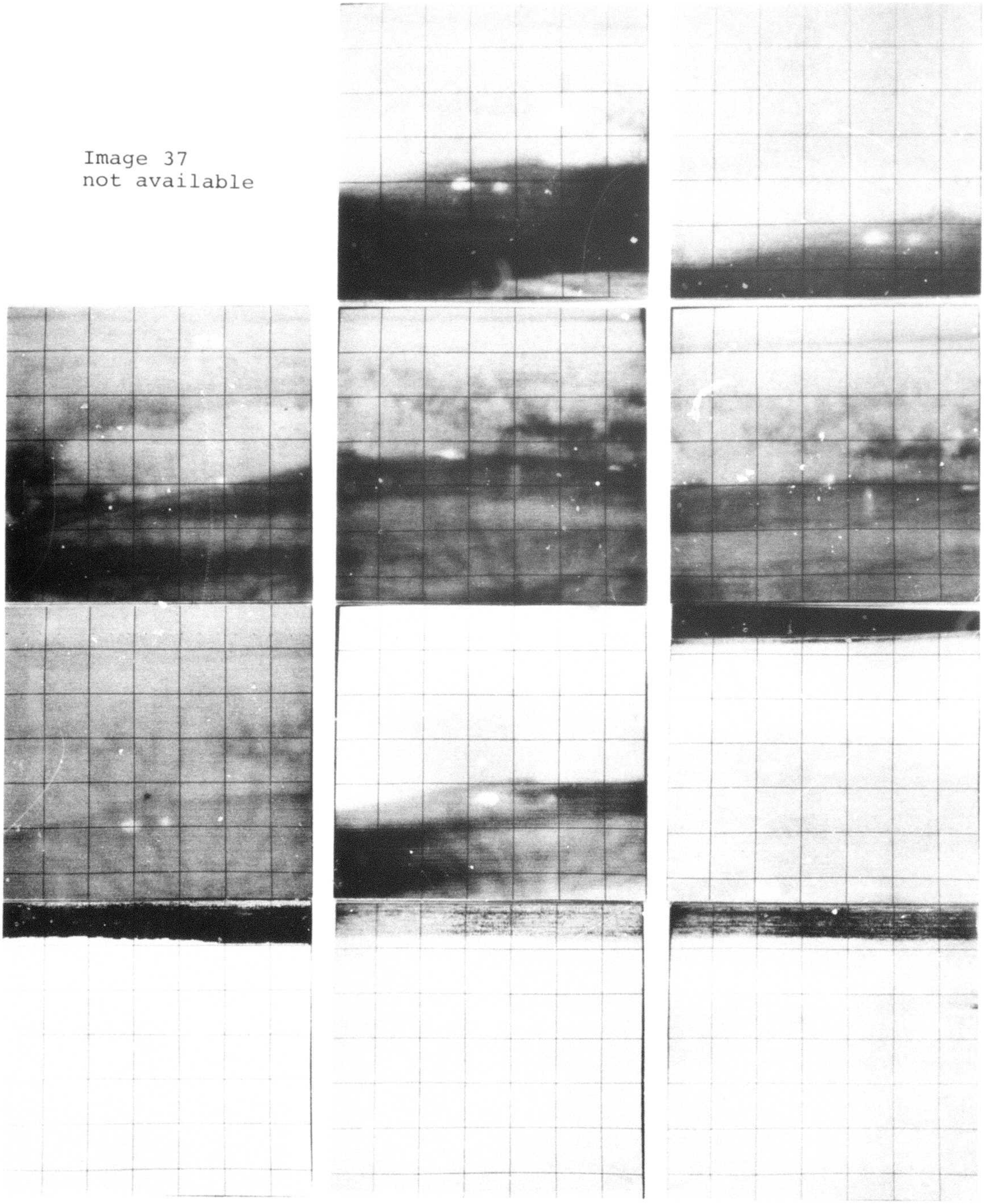


Figure 8.1b (continued). Images 37-48.

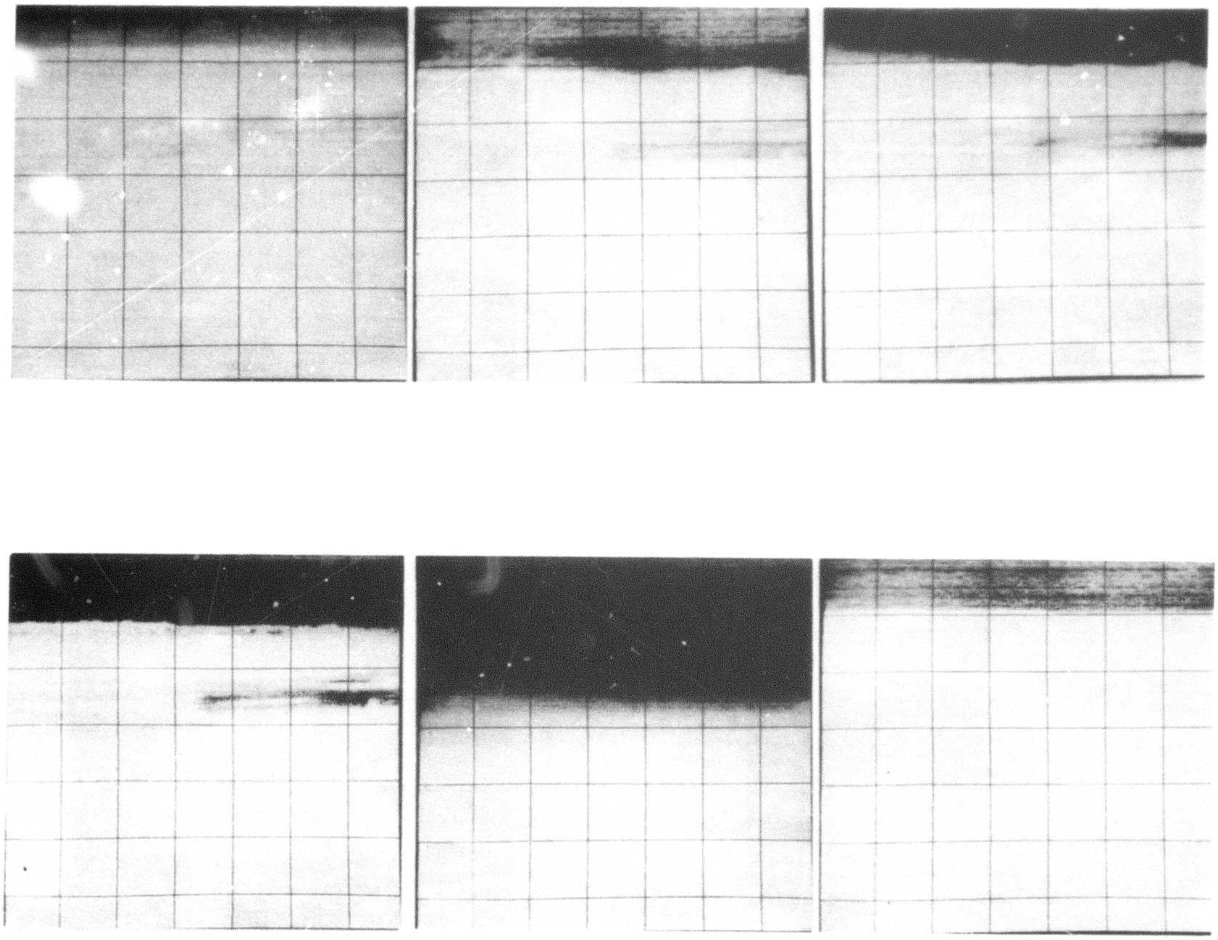
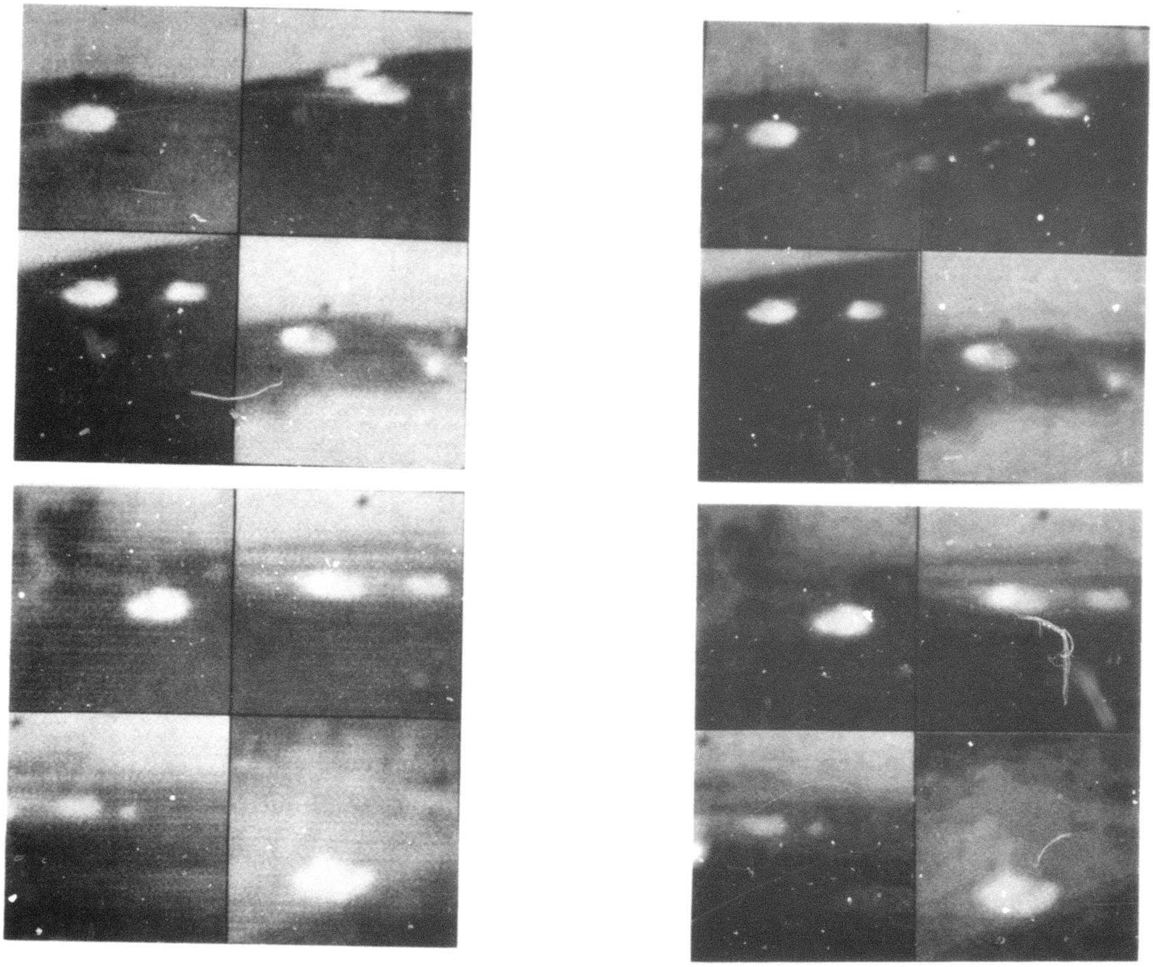


Figure 8.1b (continued). Images 49-54.



Original windows.

5x5 median filtered windows.

Figure 8.2. Examples of false contour effect from median filtering the Alabama data base. (windows from images 1-7).

8.3 The Sequential Data Base

A third data base (NSQ) consisting of ten sequential FLIR images has been acquired from NVL. The image content and quality are similar to those of the first data base. The sequence corresponds to every other frame from the FLIR sensor over a span of 2/3 of a second. The images show a tank against a background of trees which becomes less distinct with each sequential image. Figure 8.3 shows original and median filtered versions of the windowed sequence.

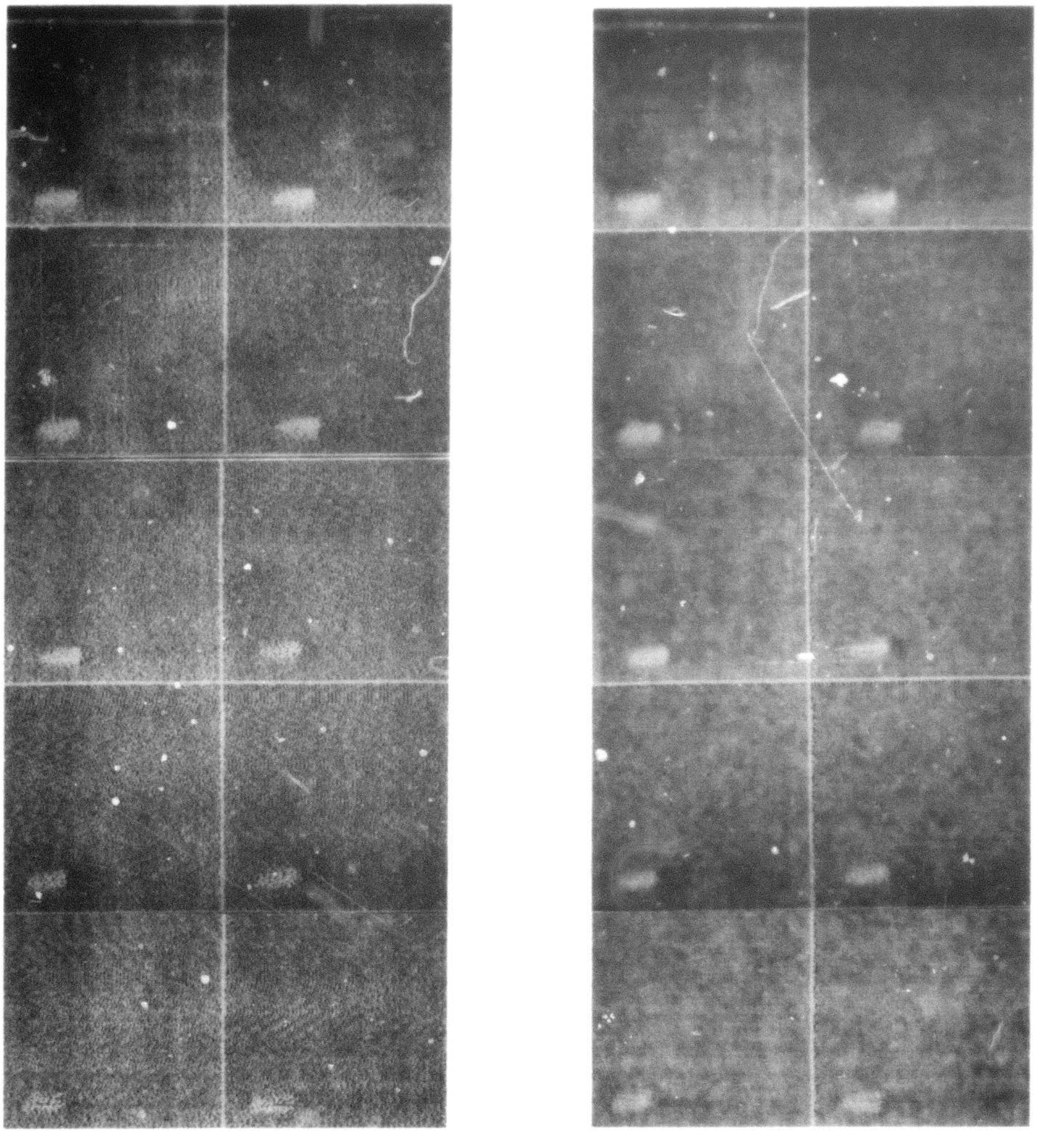


Figure 8.3. NSQ windowed data base. Ten original windows (left) and ten median filtered windows (right).

9. Plans for the Second Semiannual Reporting Period

9.1 Image Modelling

The current mathematical model has provided a rich context for algorithmic development; however, its predictive capacity is low. A more realistic model should be able to handle statistical properties of terrain type, range data, and target descriptors. In addition, a model for short term temporal variation is needed to handle frame-to-frame tracking and the iterated refinement (improvement) of the object description.

9.2 Design of a Comprehensive System

The current procedure for target cueing consists of a linear sequence of algorithmic steps. From an implementation point of view, the whole image need not be processed at each stage. Assuming a pipelined approach, we must specify the degree of parallelism, the size and location of the parallel chunks (windows) to be processed, and the internal storage requirements. In addition, since chunks are not independent (they may be adjacent in the original frame), there must be some overall control and coordination of parameter settings.

9.3 Cooperating Sources of Information

Currently, a variety of object part detectors exist for proposing object descriptors. These include edge detectors, corner detectors, and interior detectors (thresholding). The coincidence of these information sources in eliciting local responses is not only a powerful heuristic supporting the existence of the object but may also be used to extract more accurate descriptive features. Much recent work at the Computer Vision Laboratory has centered on this problem through the use of relaxation networks. The utility of the relaxation technique should be evaluated for target cueing, and modifications or alternatives suggested.

9.4 Object Extraction

We have assumed heretofore that a single threshold suffices for segmentation of a window, but this assumption is not realistic. A number of alternate approaches suggest themselves. First, the process of threshold selection can be iterated by taking the predicted threshold and extracting those compact regions which satisfy classification criteria; now, ignoring the thresholded objects, recompute the 2-D histogram and rethreshold. A second approach is to threshold the image at successively lower gray level values; extract connected components at each stage; and discard those components whose borders don't match the local maximum edge points closely. The set of components extracted can be partially ordered by containment and constitutes a tree data structure. A warm object containing hot spots would then correspond to a subtree within the data structure. Thus type of structure would be available both for object description and tracking, and for image context utilization. These approaches are more complex than those now being studied, but they are implementable (in principle) using parallel pipelined hardware, and they serve to illustrate the range of possibilities that could be employed in a "smarter" sensor.

9.5 Feature Evaluation and Target Classification

Object classification will be based on the following five classes: small target, tank, truck, APC, noise. Carefully designed tests should indicate what features provide discrimination and what additional features are needed. In a larger context, target cueing must also be responsive to situation data as well as to derived image measures. As a simple example, range data would be very helpful in target classification.

A comparative test is planned in which Westinghouse's and Maryland's cueing algorithms are applied to identical test data sets. This type of benchmark should help in evaluating the strengths and weaknesses of our method.

It is also planned to analyze the error rates for entire frames rather than just for windows. In particular, the estimation of false alarm rates will be an important goal. It should be realized, however, that it will be difficult to obtain reliable estimates of these using data bases of limited size and diversity.

9.6 Hardware Constraints

It has been pointed out that the use of 2-to-1 interlace in the sensor system presents implementation problems at the CCD level. The amount of stored data at the focal plane is thus a crucial parameter. This and other technical issues will be aired and their impact on the project assessed.

10. References

1. Algorithms and hardware technology for image recognition, Quarterly DARPA report for the period 1 May - 31 July, 1976, Computer Science Center, University of Maryland, College Park, Maryland.

2. A. Rosenfeld and A. C. Kak, Digital Picture Processing, Academic Press, 1976.

3. D. P. Panda and A. C. Kak, Image enhancement and restoration, Tech. Rep. No. TR-EE 76-17, Purdue University, W. Lafayette, Indiana.

4. N. E. Nahi and A. Habibi, Decision-directed recursive image enhancement, IEEE Trans. Ckts. and Systs., vol. CAS-22, pp. 286-293, March 1975.

5. S. Watanabe and the CYBEST group, An automated cancer pre-screening apparatus: CYBEST, Toshiba Research and Development Center, Kawasaki, Japan.

6. J. S. Weszka, J. A. Verson, and A. Rosenfeld, "Threshold selection techniques, 2", Technical Report 260, Computer Science Center, University of Maryland, College Park, Maryland, August 1973.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

9

4. TITLE (and Subtitle)

6 Algorithms and Hardware Technology for Image Recognition.

5. TYPE OF REPORT & PERIOD COVERED

Semi-annual rept.
1 May - 31 Oct[redacted] 1976

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)

10 Azriel/Rosenfeld [redacted] David/Milgram

8. CONTRACT OR GRANT NUMBER(s)

15 DAAG53-76C-0138,
DARPA Order-3206

9. PERFORMING ORGANIZATION NAME AND ADDRESS

Computer Science Ctr.
Univ. of Maryland
College Park, MD 20742

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS

U. S. Army Night Vision Lab.
AMSEL-NV-VI
Ft. Belvoir, VA 22060

12. REPORT DATE

11 31 Oct[redacted]76

13. NUMBER OF PAGES

130

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

12 129p.

15. SECURITY CLASS. (of this report)

Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Image understanding
Image processing
Pattern recognition
Target detection
FLIR imagery

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Techniques for detecting tactical targets on Forward-Looking Infrared (FLIR) imagery are being investigated. The principal topics covered include target and background models, object extraction and classification, and hardware technology applicable to real-time implementation.

UNCLASSIFIED

LB

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

403018